EasyChair Preprint

№ 9047

Blockchain Web Application Design for Legal Companies in Indonesia

Kiagus Al Amin and Yohan Suryanto

October 18, 2022

BLOCKCHAIN WEB APPLICATION DESIGN FOR LEGAL COMPANIES IN INDONESIA

[1] Kgs. Al Amin, [2] Yohan Suryanto
Faculty of Engineering, Department of Electrical Engineering
Master of Computer Engineering
Universitas Indonesia
Indonesia
Email: [1]kgs.al@ui .ac.id
Contact: [1]+6281230340121.

Abstract: *Blockchain was born from the idea to solve the problems posed by the need not to use third parties in a transaction. Issues related to the relationship between blockchain and legal firms are important issues. a carefully designed and closed system using open source applications like Angular and Firebase where later only operator privileged applications are allowed to make entries. public authorities and corporate lawyers to devote some of their time to getting to know this blockchain technology, at least as the future of law firms. As a lawyer, you are required to be able to change the conventional system with the concept of smart contracts being adopted into the blockchain system. A challenge in the development of a legal firm system where one must choose the best consensus according to needs. Apart from that, system design must also consider the performance and security aspects of the system itself. So we need a system with adequate security (maintaining confidentiality), easy to operate, and good application performance.*
Key word : blockchain, consensus, angular, firebase, law firm

1. INTRODUCTION

A technology called Blockchain that emerged a decade ago could revolutionize the global economic system [1]. Blockchain was born from the idea to solve the problems posed by the need not to use third parties in a transaction [2][3][4][5][14]. Banking and insurance systems are not the only ones affected but legal and political systems can also be incorporated into blockchain systems [6][7].

Issues related to the relationship between blockchain and legal firms are important issues[17]. It is a carefully designed and closed system in which only privileged operators are allowed to make entries[15]. This aims to maintain confidentiality and provide better service to each client's data which will improve the system created (CLAUDIA ANTAL and friends, April 2021) on the blockchain system, information on the status of COVID-19 medicines can be accessed quickly and accurately.

Distributed Ledger Technologies (DLT) is considered as a high potential platform solution that can replace traditional business models by reengineering the network and enforcing business logic and system wide data integrity [11]. This is why public authorities and corporate lawyers are devoting some of their time to getting to know this blockchain technology, at least as the future of law firms [17][18]. Their respective roles and jobs tend to change radically. In particular, they must be able to adapt to a new skill: technological expertise, in terms of computer code[22][23].

As a lawyer, you are required to be able to change the conventional system with the concept of smart contracts that can be adopted into the blockchain system[2][12]. Lawyers just have to be able to code or at least write a syllogistic specification that explains the intent of the contract. Lawyers in the blockchain world must also be a technologist. Creating a smart contract requires a new team, consisting of [18]:

- Lawyers who can design syllogistics and who have legal knowledge of contract structure and enforcement
- Technicians who provide software engineering to implement legal constructs into intelligent, self-contained transactional structures using blockchain

Project Propose

The purpose of this research is to build a Blockchain application system that overcomes the problems in the traditional system of legal firms and compare it with the blockchain system created by Claudia Antal in the IEEE journal, April 2021. Where in blockchain systems, which generally use a lot of storage systems. In this design, the author designed a blockchain system that is lossy (without a waiting room). This is because the system only works at a capacity that is not too dense.

Therefore, this becomes a challenge in developing a legal firm system where the author must choose the best consensus according to his needs. The author chose a consensus proof of authority / identity (Proof of Authority / Identity = PoA). Where the challenges that we will fix are:

- Eliminates dependence on the assumption that the current validation node has not been compromised
- Reduces the possibility of a centralized point of failure in the system

Tabel. 1.2 Matriks Perbandingan Konsensus

Name	Goals	Kelebihan	Kekurangan	Domains	implementasi
Proof of work (PoW)	Aims to deter to issue blocks in the form of computationally difficult puzzles to solve to allow transactions between untrusted participants.	• Difficult to perform denial of service by flooding the network with bad blocks. • Open to anyone with the hardware to solve the puzzle.	• Computing intensive (by design), power consumption, hardware arms race. • 51% attack potential by acquiring sufficient computing power.	Permissionless cryptocurrencies	Bitcoin, Ethereum, dan banyak lagi
Proof of stake (PoS)	aims to enable a less computationally intensive barrier to publish blocks, but still allow transactions between untrusted participants.	• Less computationally intensive than PoW. • Open to anyone who wants to transact cryptocurrency. • Stakeholders control the system.	• Stakeholders control the system. • Nothing prevents the formation of a stakeholder pool to create a centralized power. • 51% attack potential by gaining sufficient financial strength.	Permissionless cryptocurrencies	Ethereum Casper, Krypton
Delegated PoS	aims to enable a more efficient consensus model through a 'fluid democracy' where participants choose to vote and revoke delegate rights to validate and secure the blockchain.	• Economically elected delegates are encouraged to remain honest. • More computationally efficient than PoW	• Less node diversity than pure PoW or PoS consensus implementations • Greater security risk for node compromise due to restricted set of operating nodes • Since all delegates are 'know', there may be incentives for block producers to collude and accept bribes, jeopardizing system security	Permissionless cryptocurrencies Permissioned Systems	Bitshares, Steem, Cardano, EOS
Round Robin	Provides a system for publishing blocks between approved/trusted publishing nodes	• Low computing power. • easy to understand.	Requires a lot of trust between publishing nodes	Sistem yang berizin (Permissioned Systems)	MultiChain
Proof of Authority / Identity	To create a centralized consensus process to minimize block generation and confirmation rates	• Fast confirmation time • Enables dynamic block production rates • Can be used in sidechains to blockchain networks using other consensus models	• Relies on the assumption that the current validation node has not been compromised • Leads to a centralized point of failure • The reputation of a particular node carries a high potential risk as it can be compromised at any time.	Permissioned Systems Hybrid System	Ethereum Kovan testnet, POA Chain, various permissioned systems using Parity
Proof of Elapsed Time (PoET)	Aims to enable a more economical consensus model for blockchain networks, at the expense of the deeper security guarantees associated with PoW.	Computationally cheaper than PoW	• Hardware requirements to gain time. • Assumed hardware clock used to get uninterrupted time • Given the ultimate speed latency limit, true time synchronicity is essentially impossible in a distributed system [13]	Permissioned Networks	Hyperledger Sawtooth

2. BASIC THEORETICAL

2.1 Business Classification

The business classification that will be applied is very important, where in general business is in the form of synchronized data exchange. Designing a blockchain with a service platform using a Microservice architecture takes the same approach for independent services [21]. we focus on the service constraint on the business constraint of a service, aiming to clarify where the code is for a particular function. According to Melanie Swan by keeping the service focused on explicit boundaries, in order to avoid growing too big, with all the associated difficulties it can cause [7].

In fig.2.1 is a business process flow diagram of the service company. The variables obtained are in the form of client data, client documents, company documents related to the client, and information from the court regarding the client's case [8]. This will be the basis for the author to build an appropriate API (Application Programming Interface).

2.2 System Blockchain

Blockchain is a record-keeping system for transactions or exchanges of value (not just money!) in a peer-to-peer manner. This means that there is no need for trusted intermediaries such as banks, brokers, or other services to become a trusted third party [24]. The legal industry will also be transformed and enter the digital blockchain technology and related scripting languages and protocols known as smart contracts [1][3][7][24]. Lawyers need to understand the concept of communicating securely and protecting their clients' data. In particular, they need to understand blockchain and smart contracts.

fig. 2. Law firm business flow process as the basic for API design

Blockchain is a record-keeping system for transactions or exchanges of value (not just money!) in a peer-to-peer manner. This means that there is no need for trusted intermediaries such as banks, brokers, or other services to become a trusted third party [24]. The legal industry will also be transformed and enter the digital blockchain technology and related scripting languages and protocols known as smart contracts [1][3][7][24]. Lawyers need to understand the concept of communicating securely and protecting their clients' data. In particular, they need to understand blockchain and smart contracts.

In general, there are two types of blockchain networks — public and private. Both are P2P networks, where a ledger is distributed among those who can participate in transactions. A copy of the ledger is replicated among participants, and parties who can make only additional transactions to the ledger will hold a copy of the ledger and will participate to reach a consensus to add blocks to the blockchain.

The Public Blockchain is the simplest and most accessible blockchain for the public. They are usually open source, non-restrictive, fully distributed, decentralized, and lacking in permission requests. Any entity with internet access can log in to the Public Blockchain to become an authorized participant and become a user, or developer. The contents of the blockchain are readily available for each node with full transparency. The main benefit of the Public Blockchain is that it cannot be controlled or does not give full authority to certain nodes. All nodes adhere to a consensus mechanism to ensure the security of the public blockchain. The most common use of Public Blockchain is for mining and cryptocurrency exchange activities[25][26][27]. Fig 2.2 represents a fully distributed Public Blockchain where every device has full access to the blockchain and can easily interact with each other.

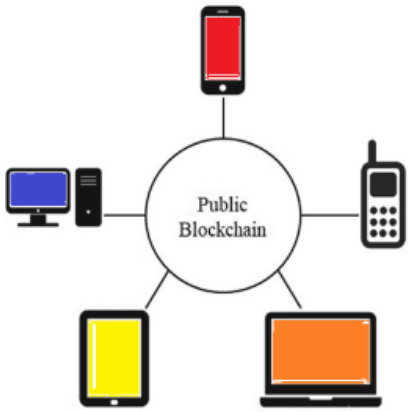


fig 2.2 A fully distributed and decentralized public blockchain

Private Blockchain is restrictive, centralized, permitted and only operates in a closed network like any organization where only selected members are allowed to participate. They have a central authority that fully controls authorization, participation, and accessibility. Participants from the same organization obligately need approval from a central authority to join the Private Blockchain. Blockchain content is only available to authorized participants and any Updates or modifications to the blockchain also require permission from the authorities. As

such, they are more secure and controllable than Public Blockchains and are commonly used in e-voting, supply chain management, etc. Hyperledger and R3 Corda are popular examples of Private Blockchain[28][29]. Fig 2.3 provides a Block Diagram of a Private Blockchain.

Hybrid blockchain is a type of blockchain technology that combines elements of a private and public blockchain. This allows organizations to set up private permission-based systems in addition to public permissionless systems, allowing them to control who can access certain data stored on the blockchain, and what data is made public. Typically, transactions and records in hybrid blockchains are not published but can be verified when needed, such as by allowing access via smart contracts. Confidential information is stored on the network but can still be verified. Although a private entity may have a hybrid blockchain, it cannot modify transactions. When users join the hybrid blockchain, they have full access to the network. The user's identity is protected from other users, unless they are involved in a transaction. Then, their identities were revealed to the other party. Hybrid blockchains are usually used by medical record systems, real estate etc. [30][31][32][33].

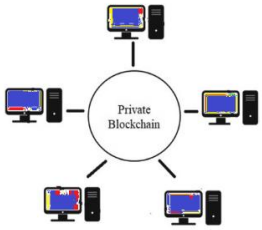


fig 2.3 Centralized Private Blockchain

2.3 Firebase

Firebase is an open source software developed by Google that allows developers to develop iOS, Android, and Web applications [48]. Firebase provides tools to track analytics, report and fix app crashes, create marketing and product experiments [48][49].

Firebase is useful for designing/producing, enhancing, and developing apps, and the tools it provides cover most of the services that developers would normally have to build themselves. This includes things like analytics, authentication, databases, configuration, file storage, push messages, and so on. The service is hosted in the cloud, and on a small scale from the developer side [50] as fig 2.4.

fig 2.4 Firebase main service

2.4 Web Stress Measurement

Measurement of the maximum load on the web design (Stress Test) by increasing the number of virtual users until it reaches the desired load. Stress testing is the process of placing the system under extreme conditions to verify the robustness of the system and/or to detect various load related problems (eg memory leaks and deadlocks). Examples of such conditions can be related to load (putting the system under normal or extreme heavy loads [28], [29], [30], [36], [37], [38], limited computing resources (e.g., CPU high [78]), or failure (eg, database failure [20]).In other cases, stress testing is used to: evaluate software design efficiency [66], [67], [68], [69].

We try to do this in order to get good results (faster), although less accurate, by combining load testing with an analytical performance model or simulation that we will create using the Law of Response Time [13][14][15]], is formulated as follows:

N_{vu} = Number of Virtual Users
N_C = Number of concurrent requests being processed
X_0 = Average throughput, number of requests per second
R = Average response time per request, in second
Z = Average processing time by the system, in second

(1) $$R = \frac{N_{vu}}{X_0} - Z$$

Virtual user is a user who accesses the network or application that we create, where the user is generated or created through a load generator. Here the author uses the load generator provided by firebase by calling it programmatically.

Web throughput is a function of load rate — where is the N_C number of concurrently executing requests. We define a service request as Di where the value of i is the total average time it takes the request to receive the service from the source [15]. So formulated:

(2) $$X_0(N_C) = f\ (D_1, \dots, D_{K,}\ N_C)$$

to show that throughput is a function of load levels and service demands on the Web site's K resources. Since the same is true for response times, we can write that:

(3) $$R(N_C) = g\ (D_1, \dots, D_{K,}\ N_C)$$

Then, combine equations 1 and 3, until we get:

(4) $$N_{vu} = [\ R(N_C) + Z\ ] \times X_0(N_C)$$

We can now use analytical or simulation models to predict response times and throughput with different values of N_C load, and use equation 4 to estimate the number of virtual users we need to generate a given $N_C$value.

2.5 Proof of Authority / Identity Consensus

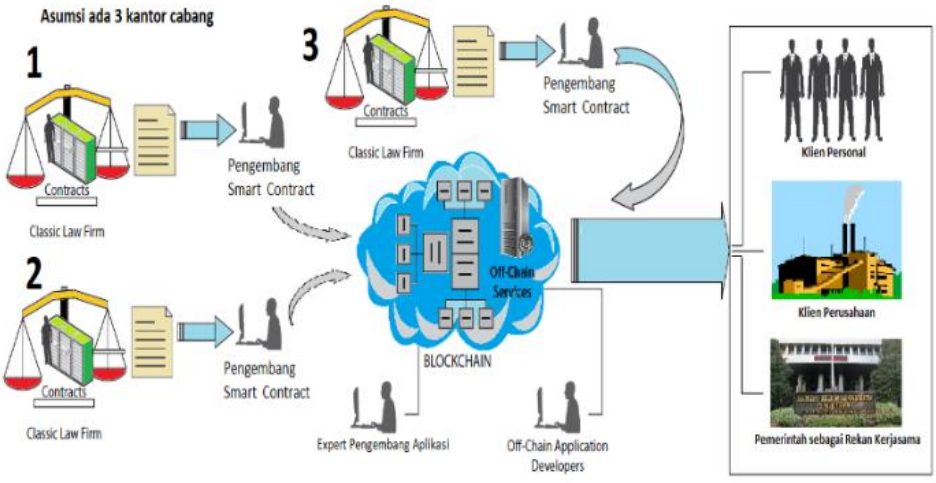
Consensus means mutual agreement, so PoA (Proof of Authority / Identity) consensus means mutual agreement on the authority of the authorities in the system. For the design of the blockchain system itself, PoA is almost the same as the Hybrid system. Where this is a merger between two consensus, namely PoW (Proof of Work) and PoS (Proof of Stake) [51]. In this design, the PoA consensus is the consensus that best fits the actual needs of the law firm.

PoA consensus design itself cannot be separated from the CAP theorem. The CAP (Consistency, Availability, and Partition Tolerance) theorem is one of the understandings that we must have when designing distributed systems or more specifically microservice architectures [52].

3. Design System

Figure 3.1 represents the business network design according to the blockchain system. Therefore, smart contract developers and blockchain systems are needed so that all parties, especially clients, can directly access all documents related to them. Figure 3.1b shows a new blockchain system that will replace the classic system, where it can be seen that there are 3 servers (3 attorney's offices) that will make changes from hardcopy to softcopy and validate payments by clients, all done by smart contract developers who are on each server. the office. After all the information is converted into digital, then the data is entered into the cloud database by the smart contract developer. The data entered by the smart contract developer is gradually in accordance with the process of the justice system in Indonesia.

In Figure 3.1 there is also an Application Developer Expert who functions as a web application developer and checks for threads (threats) on the system. Then there are off-chain application developers who will conduct research and development of blockchain systems according to developments and demands.



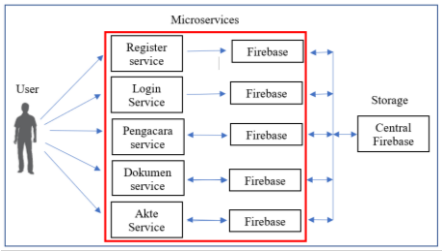
Gambar 3.1 Blockchain Business Network

3.1. Usecase System Design

Due to the frequent and rapid changes in software requirements, microservices architecture has become the dominant pattern since 2010 [46][47]. The microservices approach Figure 3.3 consists of freely bundled services that provide advantages in agility, scalability, and resiliency of development [45][46]. In addition, they offer the

ability to isolate the work of disparate teams from one another, reusing services across multiple projects and scale development organizations[46].

Based on Figure 3.1 where the use case of the Law Firm business process uses a microservice architecture on the front-end on each server that will be designed by the author as shown in Figure 3.2 below:



Gbr 3.2 Usecase Microservice arhitecture

In Figure 3.2 where the user here is defined into 3 types of users, such as:

1) Clients, namely users who come from the lawfirm's clients who have the authorization to write and read only the contents of the profile provided on their own account. This user cannot view other client's account data, nor can they delete the contents of their own account.
2) Lawyers, namely work partners and employees who work as lawyers who have been registered in the database. This type of user has authorization in the form of only being able to view and read information from each account that has been handled and cannot delete the data.
3) Administrator, this type of user has unlimited authorization for all data or information contained therein.

3.2. Blockchain system design

The blockchain system that will be designed starting from the first step of this process is registering and validating the client's identity before accessing the document information (see Figure 3.3). This is done by the system via an account code which contains a blockchain transaction hash, a personal identification number hash, and a secret key hash. Using the extracted hash, Lawyer performs on-chain identity verification. Using the hash of the two values compared to the root stored on the blockchain during the beneficiary registration step.

Overall, users are divided into administrators, lawyers, and clients, where these three users have different authorizations, including page access in the application. In web design using the typescript language angular application. In Figure 3.3 is the initial display on the screen, which only shows home, employee, register and login.

The home and employee sections are pages that can be accessed by everyone including those who have not registered, this aims to promote and show clients or prospective clients the experience and capabilities of the legal company as a reference for clients and prospective clients who need service assistance. lawyer.

Then the register and login pages are input for the blockchain system which will be entered into the database and used as authentication for each user. Here the author only uses email and password as authentication when logging in. At that time, the authorization system began to be implemented according to the type of users, namely administrators, lawyers, and clients. It can be seen in Figure 3.4 is a flowchart of checking whether there is an addition/deletion of case data.

However, if the Client is not registered in the chain, it can also be created by the administrator (smart contract developer). Just like banking, to make services easier. If needed, the administrator can also provide brief training on how to use it.

3.3. Optimization Service Architecture

The first step of the client to send optimization, see Figure 3.4. It attaches the required data as the request payload. This data can be of various formats and sizes, depending on the use case. This can be a reference to existing data (such as a scenario identifier), the user configuration of the optimization, the entire instance for the optimization, or a combination of the three. Each option has different advantages and disadvantages and trade-offs, such as payload size, availability to clients and accessibility to internal identifiers.

An optimization service requires the following components, while managers and workers are the main components of OaaS; messaging systems, databases and object storage enable external infrastructure components:

- Manager, provides external client access to optimization services. It provides the features presented in the following sections for clients interacting with the service: submit optimizations, retrieve state and solutions, and stop optimizations.
- Worker, each instance is responsible for completing one mathematical optimization at a time.
- Database Stores metadata for optimization requests, such as status changes, identifiers, or delivery information.

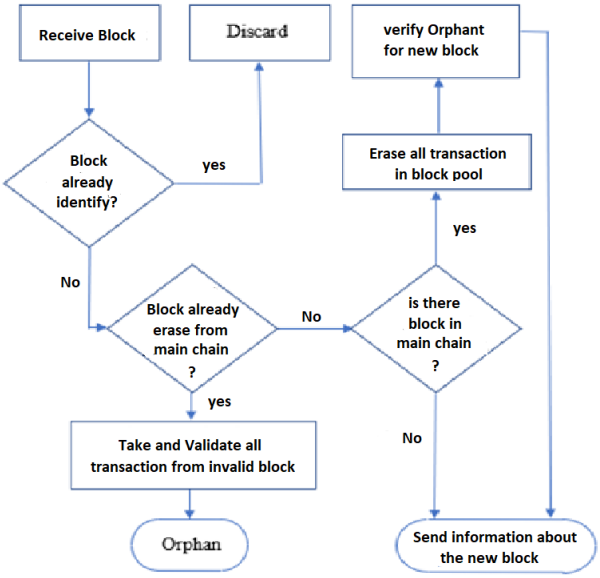


fig 3.3 Transaction flowchart add/remove a block

3.4. Web Stress Measurement Design

We will enter the webstress design into the system which can be seen only by the administrator. This is so that the administrator can see the usage load. By adding a program and viewing it through chrome by pressing the F12 key, the operator can estimate the density of users who access the site.

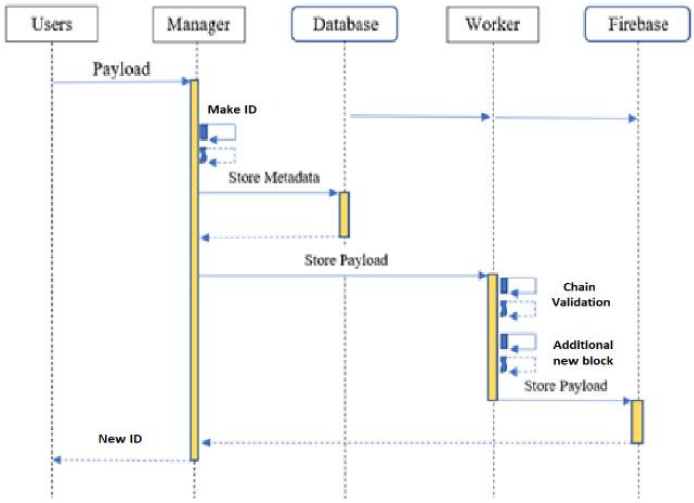


fig 3.5 UML data transfer sequence optimization

4. SYSTEM ANALYSIS

4.1. Analyzing the Design Result Blockchain System

Based on the blockchain design made by Claudia in her thesis, there are differences in the use of a waiting room (Queuing) where the function of the waiting room in the system built by Claudia is for public services. Meanwhile, the system designed for the lawfirm system does not use a waiting room for users who access it due to limited users.

User restrictions for those who access the system have several purposes and reasons that underlie the design of the system, such as:

a. Cause the number of visitors and users is not too crowded as in business or trading systems in general.
b. Due to maintain the stability and performance of the system in order to provide services with maximum speed.
c. Become a separate security system against Ddos attacks, which can cause the system to crash
d. Provide assurance that data or information from each client is safe and easy to access.

The following in Figure 4.1 describes the sequence procedure that occurs as a whole in the blockchain system created regarding legal companies and the actual conditions that occur.

In step 7 in Fig 4.1 is an off chain condition, at this stage the client agrees and signs an agreement letter to appoint a legal representative who will represent or assist in the trial of his case. After that, the client tells the chronology of the case he is facing and completes the supporting documents and evidence in his case. Then complete the payment according to the agreement to the attorney and administration at the office.

In step 12 the lawyer will validate all the chronology of his client's case according to their respective systems so that they can be submitted in court. Apart from that, the admin will also validate the payments made by the client. After being validated then all information and proof of deposit will be entered into the client's block. If there are changes or additions during the trial process in the contents of the block, only the administrator (smart contract developer) can do it. This is like a banking system that hands over the power to input banking transactions to a teller.

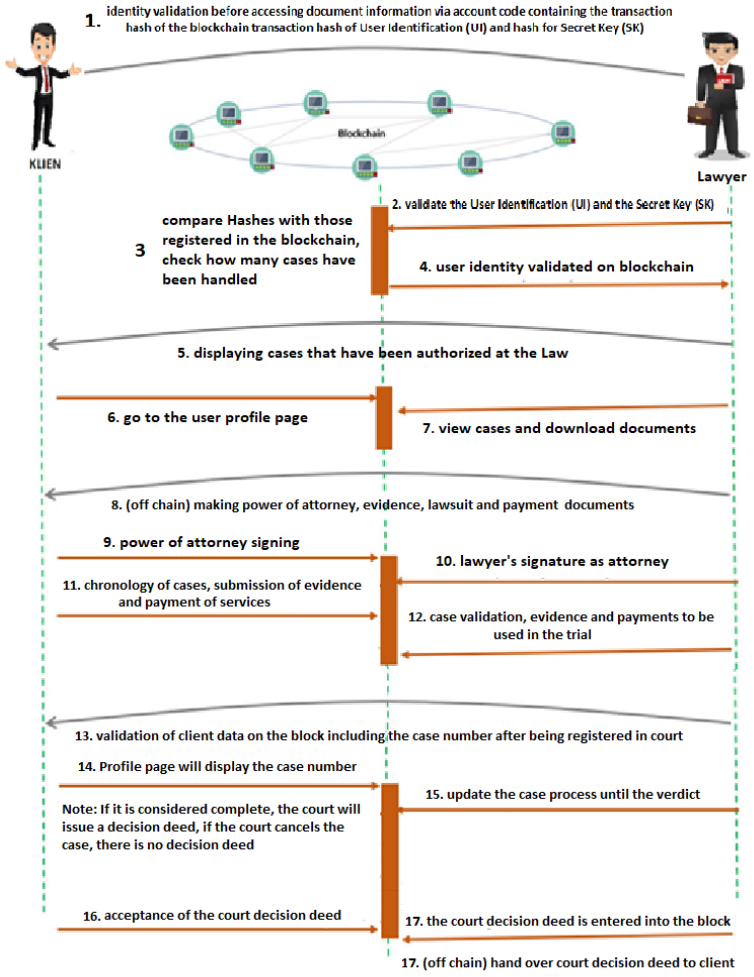


fig 4.1 Law firm business flow Diagram

4.2. Analyze performance using stress tests

From the equation (1) above, several results will be obtained as improvement material for the design system. As in Fig 4.2 which is a response time graph for virtual users. we use the features that have been provided in firebase and just make a call using the scrypt program. To see the results, the authors observed using chrome analysis by pressing the f12 key. Although not too detailed, but the results obtained are in accordance with the expected.

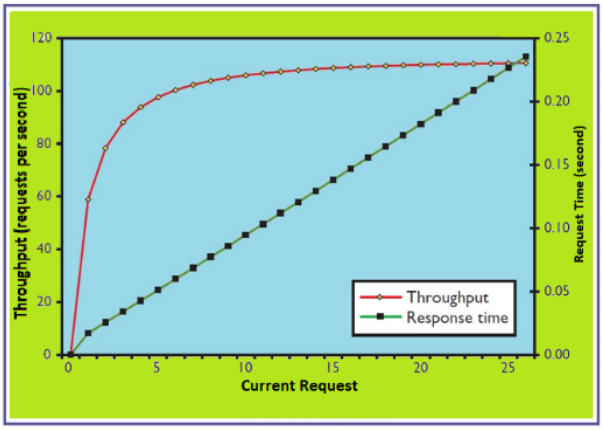


fig 4.2 Performance against Load (Virtual User)

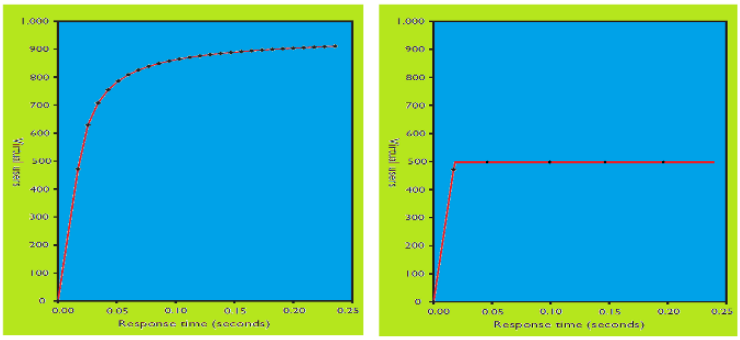
The load testing system is quite useful here. They can, for example, generate scripts for multiple virtual users to measure service requests, which are independent of the load nor can we limit the load to get the desired result. You can then use the service request as an input parameter for the performance model. Consider, for example, a Web site that has a service request for processing and I/O of 8 and 9 milliseconds, respectively. Using the Average Value Analysis method, you can calculate Xo(Nc) and R(Nc) for various values of Nc. Seen in the table of experimental results 4.1 below.

Table 4.1 Performance test current request vs virtual user

Current request (Nc)	Throughput (Xo)	Response Time Rate (R)
5	22,216	0,048 second
10	44,432	0,096 second
15	66,648	0,144 second
20	88,864	0,192 second
25	111,086	0,24 second

Table 4.1 describes the current request or the number of requests that are processed simultaneously which increases so that the Throughput value increases along with the increase in the average value of the system response. This is very influential on system performance if the user access is not restricted which can be fatal to the system.

Figure 4.2 shows the resulting curves of R(Nc) and Xo(Nc) versus Nc. The figure also shows that for Nc = 19, the number of virtual users — calculated using equation 4 and assuming an average runtime of 8 seconds — will be 897. Figure 4.3(a) shows the variation in response time as virtual users increase: As the number approaches 800, response time increases rapidly.



(a) (b)

fig 4.3 (a) Normal Respons Times vs Virtual User
(b) Respons Times vs Virtual User with limitation

Use an analytical or simulation model to predict response times and throughput for different values of the load level Nc, and use equation 4 to estimate the number of virtual users we need to generate a given value of Nc. So, in our example, the maximum throughput is 1/0.009 = 111.1 requests per second, which is the horizontal asymptote of the throughput curve of Figure 4.2.

In the graphic image 4.3(b) this is the result of modifications made to the designed software. This aims to improve the performance of the system that we make. This is to maintain the stability of the response time so that

it can serve every incoming user with a speed of less than 5 seconds for each request process. Where if we look at the maximum throughput is 1 / 0.009 = 111.1 per second, then if 500 users access our application at the same time it will take 4.5 seconds, regardless of the network used and the type of device accessing it. The results and objectives obtained from the restrictions from users are:

- ✓ The reason why restrictions are placed on the system is because it is predicted that this system will have fewer users accessing it. This application is not finance (related to finance) nor e-commerce (sales and purchases transactions / product trade). This application is only to provide information regarding clients who use the services or visitors who will seek the services of the legal company.
- ✓ The system is faster in completing services for its users where the average processing speed is less than 5 seconds per user, thus giving the impression of elegance and high economic prestige. Although using open source applications from firebase, the quality of the system is like a paid system because it has a high service speed.
- ✓ Can be used as security, where the system becomes more immune or resistant to thread (threats) attacks from hackers such as DDos attacks. Although the system is given an attack in the form of many requests from outside, the system will automatically limit the number of users who will access this system.
- ✓ Making the system more stable and reliable even though in the design and application of this system using an open source application. Access speed is affected by the network used by the user and the type of device. The higher the device specifications and the better the network used to access, the faster the process can be completed.

The development of this application is very flexible, where data security is guaranteed and paid-based development is still a lot of features offered for the development of this application in the future.

4.3. Analyzing performance of blockchain system consensus

The performance of the consensus designed in this application is quite satisfactory. Although using an open source application, for authorization it can be connected directly with the authentication of each user. It is also simpler because there is no need to create a docker (software liaison between the application and the database system) because it is already integrated in the features offered.

Here we use authentication for each user using email because this email-based authentication is free. Meanwhile, those who pay for authentication can use a mobile number or a special password that will be generated by the system according to the design. This can be seen on the official firebase page regarding this authentication and the amount to be paid.

The consensus made by the author is divided into four when viewed from the type of users, such as;

- ✓ **Visitors,** visitors are users who only stop by and see the home page, register and login only. Where the home page is the front end that serves as a place to promote the services we offer.
- ✓ **Client (user),** client are those who already have an account in the company's database system. This type of user has authorization to view (read) more pages and download them, but cannot write, modify and delete data or information contained there. Here the author makes it into role 3 in the software.
- ✓ **Lawyers,** are workers who are directly involved in solving client problems. So it has role 2 which means it has the authorization to read, write, and download data. Role 2 does not have the authorization to change and delete data or information that is already in the firebase database.
- ✓ **Administrator,** is a user who has full access to the system. Administrators can read, download, upload, write, modify data, and delete data. Apart from that, administrators can also develop systems from existing ones according to their needs. Administrator occupies role 1 which has full access to the system.

For each role or authorization, visitors have different abilities in accessing the number of pages and read and write operations, as well as delete. This can be seen in the appendix attachment sheet. Where each role has set which pages can be accessed by its users. This will further ensure the data security of the clients of this law firm. Then it also makes it easier for lawyers to know the schedule or cases that are in process.

Here the author has implemented a Proof of Authority / Identity = PoA where each user identity has its own authority according to the status (role) of the user. From the results of blockchain designs for legal companies using PoA it has several advantages including fast confirmation times, enabling dynamic block production rates. This means that the confirmation time of a request becomes shorter, only checking the key that comes from authentication from that user. Then the block production level becomes more dynamic, meaning that the key originating from each user can be reused as authorization to open pages as well as read, write, and delete processes in the application.

5. CONCLUSION

- ✓ The law firm's blockchain system is more concerned with data confidentiality when compared to Claudia Antal's blockchain system.
- ✓ The resulting performance is also better due to maintaining high speed performance when compared to the blockchain system made by Claudia Antal

- ✓ The use of user restrictions in this application has two significant benefits, namely as an increase in application performance and as security in case of hacker attacks such as DDos, when compared to Claudia Antal's blockchain system, it is more vulnerable to attacks.
- ✓ The application of PoA consensus is the best consensus in designing a system for the scale of a Law Firm like this when compared to Claudia Antal's blockchain system which is more public.

Reference

[1] M. Anthony TCHAKERIAN. "*The Blockchain Revolution: Prospects and limitations*" Journal publication on ResearchGate, 2020. Available on: https://www.researchgate.net/publication/340435151

[2] Claudia Antal, Tudor Cioara, Marcel Antal, Ionut Anghel.. "*Blockchain Platform For COVID-19 Vaccine Supply Management*". IEEE, 2021

[3] Boris Barraud, Journal HAL. " Les blockchains et le droit " available on : https://hal.archives-ouvertes.fr/hal-01729646

[4] Dylan Yaga, Peter Mell, Nik Roby, Karen Scarfone. Journal NIST Publication. "*Blockchain Technology Overview*" NISTIR 8202, Available on : https://doi.org/10.6028/NIST.IR.8202

[5] Weilin Zheng, Zibin Zheng, Xiangping Chen, Kemian Dai, Peishan Li, Renfei Chen. "*NutBaaS: A Blockchain-as-a-Service Platform*" , IEEE Journal Publication, 2019

[6] Roman Beck, Christian Becker, Journal Dagstuhl. "*Opportunities and Risks of Blockchain Technologies*" 2017. Available on : http://www.dagstuhl.de/17132

[7] Melanie Swan, *"Blockchain Blueprint for a New Economy*" First Edition, 2015:

[8] Bikramaditya Singhal, Gautam Dhameja, Priyansu Sekhar Panda. "*Beginning Blockchain: A Beginner's Guide to Building Blockchain Solutions*" 2018. Available on : https://doi.org/10.1007/978-1-4842-3444-0

[9] P. Software and S. Units, "*What is a container ? a standardized unit of software package software into standardized units for everywhere : Linux* ," pp. 1–5, 2020

[10] Martin Fowler, *"Microservices a definition of this new architectural term".* 2021 (0nline) Tersedia: https://martinfowler.com/articles/microservices.html , diakses taaggal 19 Sept 2021 jam 01:05 WIB

[11] Beck, R. (2018). "*Beyond Bitcoin*: *The rise of blockchain world. Computer"*, 51(2), 54-58.

[12] Ravindhar Vadapalli, "*Blockchain Fundamentals Text Book: Fundamentals of Blockchain*" , Available on: https://www.researchgate.net/publication/345045424. Jurnal Publication from Researchgate. 2020

[13] P.J. Denning and J.P. Buzen, "*The Operational Analysis of Queuing Network Models*," *ACM Computing Surveys*, vol. 10, no. 3, Sept. 1978, pp. 225-261.

[14] Zheng, Z., Dai, H., & Xie, S. (2018). "*Blockchain chal-lenges and opportunities: A survey*", International Journal of Web and Grid Services, 14 (4), p. 352-375.

[15] Trew, C., Brandon, G., & Dorier, N. (2019). "*Stratis Blockchain Solutions whitepaper*", Stratis Group Ltd., p. 1-20. https://stratisplatform.com/files/Stratis_Whitepaper.pdf

[16] S. Daya et al. (2015, August). "*Microservices from Theory to Practice: Creating Applications in IBM Bluemix Using the Microservices Approach*", (1st edition). [On-Line]. Available: http://www.redbooks.ibm.com/abstracts/sg248275.html [Agustus 15, 2021]

[17] Marçal Sintes-Olivella; Enric Xicoy-Comas; Elena Yeste-Piquer. (2020). "*Blockchain at the service of quality journalism: the Civil case*" Journal Researchgate. https://www.researchgate.net/publication/345253347

[18] Joseph J. Bambara, Paul R. Allen. " *Blockchain : A Practical Guide to Developing Business, Law, and Technology Solutions* " Published by McGraw-Hill Education, 2018

[19] Asikin, Z, dkk, "*Pengantar Hukum Perusahaan"*, Prenadamedia Group, Jakarta, 2016.

[20] Livia V. Pelle, 2012. "*Peranan Etika Profesi Hukum Terhadap Upaya Penegakan Hukum Di Indonesia*" jurnal *Lex Crimen* Vol.I/No.3/Jul-Sep/2012 https://media.neliti.com/media/publications/3153-ID-peranan-etika-profesi-hukum-terhadap-upaya-penegakan-hukum-di-indonesia.pdf

[21] Sam Newman. *Building microservices designing fine-grained systems*, P-2

[22] Piotr Karwatka, Mariusz Gil, Mike Grabowski, Aleksander Graf, Pawet Jedrzejewski, Michał Kurzeja, Antoni Orfin, Bartosz Picho. *"Microservices Architecture for eCommerce"* . 2020

[23] James Hartman. "*Typescript vs JavaScript: What's the Difference?* " [On-Line]. Available: https://www.guru99.com/typescript-vs-javascript.html [September 09, 2021]

[23] Firebase official website. "*Documentation*" [On-Line]. Available: https://firebase.google.com/docs/reference [August 18, 2021]

[24] Pratyusa Mukherjee, Chittaranjan Pradhan. Journal. "*Blockchain 1.0 to Blockchain 4.0—The Evolutionary Transformation of Blockchain Technology*". Researchgate Journal Pu lication, May 2021.

[25] Zheng, Z., Xie, S., Dai, H.N., Chen, X., Wang, H.: "*Blockchain challenges and opportunities: a survey*". Int. J. Web Grid Serv. **14**(4), 352–375 (2018)

[26] Norman,M.D., Karavas, Y.G., Reed, H.: "*The emergence of trust and value in public blockchain networks*". In: IX International Conference on Complex Systems, p. 22 (2018)

[27] Toshendra Kumar Sharma "Public VS Private Blockchain : A Comprehensive Comparison" Accessed : 10 Sept 2021, on: https://www.blockchain-council.org/blockchain/public-vs-private-blockchain-a-comprehensive-comparison/

[28] Pongnumkul, S., Siripanpornchana, C., Thajchayapong, S.: Performance analysis of private blockchain platforms in varying workloads. In: 2017 26th International Conference on Computer Communication and Networks (ICCCN), pp. 1–6. IEEE (2017) and also available on researchgate on : https://www.researchgate.net/publication/319889164_Performance_Analysis_of_Private_Blockchain_Platforms_in_Varying_Workloads

[29] Euromoney learning team. "The rise of private blockchains". Accessed: 26 Sept 2021. On: https://www.euromoney.com/learning/blockchain-explained/the-rise-of-private-blockchains

[30] Hazem Marar, Rosana Marar. " *Hybrid Blockchain*" Researchgate journal, December 2020. On : https://www.researchgate.net/publication/345934134_Hybrid_Blockchain

[31] Julien Polge, Sankalp Ghatpande, Sylvain Kubler, Jérémy Robert, Yves Le Traon. "*BlockPerf: A hybrid blockchain emulator/simulator framework*", IEEE Journal Publication, 2017. DOI : 10.1109/ACCESS.2021.3101044. 2021

[32] Ateniese, G., Chiaramonte, M.T., Treat, D., Magri, B., Venturi, D.: U.S. Patent No. 9,959,065. U.S. Patent and Trademark Office, Washington, DC (2018)

[33] Manian, Z.N., Krishnan, R., Sriram, S.: U.S. Patent Application No. 15/212,018 (2017)

[34] https://searchcio.techtarget.com/feature/What-are-the-4-different-types-of-blockchain-technology , accessed : 28/9/2021

[35] https://openledger.info/insights/consortium-blockchains/ , accessed : 28/9/2021

[36] https://blockchain.intellectsoft.net/blog/how-the-consortium-blockchain-works/ , accessed : 28/9/2021

[37] Xuefeng Liu, Gansen Zhao, Xinming Wang, Yixing Lin, Ziheng Zhou, Hua Tang, and Bingchuan Chen. *et al*., "*MDP-Based Quantitative Analysis Framework for Proof of Authority*," 2019 International IEEE Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery *(CyberC)*, 2019, pp. 227-236. on: https://ieeexplore.ieee.org/document/8945878

[38] Vincent Gramoli, Parinya Ekparinya, and Guillaume Jourjon. "*The attack of the clones against proof-ofauthority*", researchgate publication journal, 02 2019. On: https://www.researchgate.net/publication/331397046_The_Attack_of_the_Clones_against_Proof-of-Authority

[39] Omar Hasan, L. Brunie, and E. Bertino. "*Privacy preserving reputation systems based on blockchain and other cryptographic building blocks: A survey*". 2020

[40] M. Pease, R. Shostak, and L. Lamport. "*Reaching agreement in the presence of faults*". *Journal of the ACM*, 27(2):228–234, 1980.

[41] Stefano De Angelis, Leonardo Aniello, Federico Lombardi, AndreaMargheri, and V. Sassone. "*Pbft vs proofof-authority: applying the cap theorem to permissioned blockchain*". 01 2017.

[42] Sivleen Kaur, Sheetal Chaturvedi, Aabha Sharma, and Jayaprakash Kar. "*A research survey on applications of consensus protocols in blockchain. Security and Communication Networks"*, 2021:1–22, 2021

[43] "*V3 White Paper Executive Summary*". Accessed 28/9/2021, on : https://www.storj.io/Storj-White-Paper-Executive-Summary.pdf

[44] https://www.storj.io/how-it-works

[45] Dmitry N, Manfred SS (2014) *On micro-services architecture*. International Journal of Open Information Technologies 2(9), ISSN 2307-8162.

[46] Stefan Guericke, Andrea Cassioli, " *A Framework for Mathematical Optimization in Microservice Architectures* " , on: https://www.semanticscholar.org/paper/A-Framework-for-Mathematical-Optimization-in-Guericke/2c8b8593706ef3c25e1adc280480668bc38d05d4

[47] Hoda R, Salleh N, Grundy J (2018) *The rise and evolution of agile software development*. IEEE Software35(5):58–63, on: http://dx.doi.org/10.1109/MS.2018.290111318

[48] Linda Rosencrance, Google Firebase, access on : 07/11/2021 https://searchmobilecomputing.techtarget.com/definition/Google-Firebase

[49] Google Firebase official website. access on : 07/11/2021 https://firebase.google.com/integrations?hl=id

[50] Doug Stevenson, "*What is Firebase? The complete story, abridged.*" access on : 07/11/2021, https://medium.com/firebase-developers/what-is-firebase-the-complete-story-abridged-bcc730c5f2c0

[51] Apa itu Proof of Authority (PoA) : konsensus Hybrid dari PoW dan PoS. access on : 07/11/2021, https://www.invesnesia.com/apa-itu-proof-of-activity-poa-konsensus-hybrid-dari-pow-pos/

[52] Piotr Karwatka, Mariusz Gil, Mike Grabowski, Aleksander Graf, Pawet Jedrzejewski, Michał Kurzeja, Antoni Orfin, Bartosz Picho. "Microservices Architecture for eCommerce", P- 28

[53] Situs resmi Angular. "*What is Angular*" [On-Line]. Available: https://angular.io/guide/what-is-angular [Februari 18, 2021]