

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/340173622>

Blockchain Based Warehouse Supply Chain Management using Hyperledger Fabric and Hyperledger Composer

Article in *International Journal of Information Technology* · January 2020

DOI: 10.35940/ijitee.C1033.0193520

CITATIONS

4

READS

1,011

3 authors:



Nur Hazbiy

Brawijaya University

1 PUBLICATION 4 CITATIONS

[SEE PROFILE](#)



F. Astha Ekadiyanto

University of Indonesia

25 PUBLICATIONS 56 CITATIONS

[SEE PROFILE](#)



Anak Agung Putri Ratna

University of Indonesia

72 PUBLICATIONS 347 CITATIONS

[SEE PROFILE](#)

Some of the authors of this publication are also working on these related projects:



Essay Grading in Bahasa (SIMPLE-O) [View project](#)



FANS Project [View project](#)

Blockchain Based Warehouse Supply Chain Management using Hyperledger Fabric and Hyperledger Composer

Nur Hazbiy Shaffan, F. Astha Ekadiyanto, Anak Agung Putri Ratna

Abstract: *This research investigates the operation of smart transactions in the blockchain and demonstrates the process of transforming an existing system into the blockchain using the Hyperledger Blockchain Platform. The process includes determining business model requirements and redesigning the business model to be applied to the blockchain system using Hyperledger Composer. The supply management system is the selected case study due to its complexity that can be used to demonstrate blockchain implementation. This research explored the utilization of Hyperledger Fabric as well as Hyperledger Composer technology. The implemented system is then under test for performance evaluation. According to experimental results, read access has the shortest duration needed, which is 33% faster than Create, 69% compared to Delete, and 82% compared to Update. Update function has the longest average duration needed. Furthermore, reading using query would dramatically slow down the Read function with the increase of the average read duration over 10 commodities category by 12238.9%, over 100 commodities category by 1950%, and over 1000 commodities category by 308% than reading without using a query.*

Keywords : *Hyperledger Fabric; Hyperledger Composer; Blockchain; Smart Contract.*

I. INTRODUCTION

Nowadays, advanced science and technology can influence humans' mindset in their daily lives. This development has encouraged people to try to overcome all the problems that arise around them. Especially in data processing, the role of the database is significant. Database processing is a mainstay device whose presence is needed by various institutions and companies either in terms of academics or in terms of business. Database processing not only accelerates information retrieval but can also improve the service's efficiency as well as maximizing effectiveness to clients. However, the technology used to validate the information is still manual and simple. Traditional methods of validation widely used today have very low efficiency.

The process of validating complex data with long

duration causes the database processing to be increased. The use of manual data validation also requires considerable time and effort that causes the transaction process to become long, which increases the operating costs of the transaction. Therefore, an effort is needed to find a data processing and storage system that is safe, fast, and still gives confidence in the data so that efforts to do data validation can be suppressed as possible.

The long data validation process can be overcome by using a database system that has high data control. The purpose of a high control data system is to reduce the possibility of data forgery as well as uphold data validation. Using a decentralized data storage system, the costs required for recording transactions can be reduced as small as possible.

Blockchain can be implemented to reduce problems in data efficiency. The Blockchain itself has been implemented in Bitcoin's database system [1]. This is a distributed ledger, which makes no one be able to doubt the integrity of the data [2]. Blockchain contains a sequence of blocks that are related to each other. Thus, attempts to manipulate data will be very difficult because of interconnection between each block, so it has to change every subsequent block. In the event of data forgery, the node where the data has changed will be different from another node. Therefore its node will then be invalidated by consensus when trying to initiate a new transaction.

The purpose of this research is to design a blockchain-based supply chain management system and to measure the performance of the blockchain system based on Hyperledger Composer and Hyperledger Fabric in performing Create, Read, Update, and Delete (CRUD) functions.

II. KEY CONCEPTS

A. Blockchain

Blockchain (distributed ledger) is a distributed database ledger, which increases continuously and is utilized by several entities. Blockchain can maintain the validity and correctness of data among nodes or parties that do not have the trust of each other. In general, blockchain is used as a substitute for third parties, decentralizes trusts, and provides a secure data storage system.

There are several types of Blockchain which are [3]:

- Permissionless



Revised Manuscript Received on January 10, 2020.

* Correspondence Author

Nur Hazbiy Shaffan*, Department of Electrical Engineering, Universitas Indonesia, Depok City, Indonesia. Email: nur.hazbiy@ui.ac.id

F. Astha Ekadiyanto, Department of Electrical Engineering, Universitas Indonesia, Depok City, Indonesia. Email: astha.ekadiyanto@ui.ac.id

Anak Agung Putri Ratna, Department of Electrical Engineering, Universitas Indonesia, Depok City, Indonesia. Email: ratna@eng.ui.ac.id

Blockchain: Where everyone can be a user to run a node, can add data and can participate in the blockchain system. Permissionless Blockchain is not suitable for co, especially in the supply chain because all data contained in this type of blockchain are generally transparent, therefore sensitive data will be visible to the public [4].

- **Permissioned Blockchain:** This blockchain is referred to blockchain system operated by registered users and use authorization system, such as in the consortium blockchains, to determine its users. Blockchain's authorization system can identify users that can participate in the blockchain system and can control who can initiate each transaction (such as hyperledger) [5] [6].
- **Private blockchain:** Only one user is registered in this type of blockchain. This blockchain runs on only a single trust domain.

B. Supply Chain Management

Supply Chain Management covers all stages of processing, from purchasing raw commodities to distributing finished commodities to end consumers. This chain consists of a network of multiple companies that are organized and participating in the same business. The term Supply Chain Management also includes the process of changing the goods, for example, from raw goods to finished goods [7].

Problems begin to emerge when Supply Chain Management develops into a complex system. This problem will require more and more integration processes between the various parties involved. With the increasing number of integrations occurring, the data validation process needed to check the correctness of the data will also increase. It will then result in a decrease the efficiency.

One effort to solve this problem is to change the current management system by improving the efficiency of Supply Chain Management performance with blockchain technology. However, the maturity of blockchain technology is still unknown when it is used to support the efficiency of Supply Chain Management because of the complexity of its business model and lack of research that can encourage the development blockchain within this field. Therefore, an excellent subject to be explored in the research discussed in this paper.

C. Hyperledger Fabric and Hyperledger Composer

Hyperledger Fabric is an implementation of the blockchain framework, and one of the open-source project in the Hyperledger projects. Hyperledger Fabric can restrict how users initiate transactions using *chaincodes* and ordering services. Hyperledger Fabric uses PBFT on an early version, but the latter version changed to no-Byzantine consensus protocol [8].

Chaincode is one of the features of Hyperledger Fabric for smart contracts. This feature is the logic of the business plan, which maintains the ledger in the system, control the world state (current value of attributes as a unique ledger) in the database, and follows endorsement policy. Ordering services are responsible for maintaining the consortium and

restricting who can read and write the data when the blocks are added to the distributed ledger [9].

The Hyperledger Fabric blockchain consists of nodes that make up a network. Because Hyperledger Fabric is a permissioned blockchain, all of the participating nodes in the network must have an identity. This identity is created by the membership service provider (MSP). With the identity, the members of the system can be trusted, which is the key to running a business ecosystem consisting of various kinds of participants, which ultimately requires a trusted third party to connect the ecosystem [10]. In Hyperledger Fabric, there are three phases consists of a proposal phase, a packaging phase, and a validating phase [11].

In Hyperledger Fabric, a system can still be managed with a business model that is built off trust between the users where the users in the system may well not entirely trust each other (they may, for example, be competitors in the same industry). This problem can be solved by using traditional consensus, such as the Byzantine-fault tolerant (BFT) consensus. BFT can increase the processing speed of up to 10,000 transactions per second on a broad implementation, as well as speed in writing transactions on the blockchain in half a second, even though peer-to-peer connections are very far from different continents.

Hyperledger Composer is an open development project and part of Hyperledger open source collaborative effort. Hyperledger Composer implemented to simplify the application of the blockchain. Hyperledger Composers can be integrated into a system that has been running blockchain as its data processor. The implementation of *chaincode* using Hyperledger Composer can be further improved by using the REST server that enables the existing system to be integrated with Hyperledger Composer via HTTP Request based on all the Entity, Transactions and Models that registered on the chaincode. Hyperledger Composer uses Hyperledger Fabric as its network, Hyperledger Composer can limit the user's privilege [12].

III. PROPOSED SYSTEM

When viewed from the designed scenario, supply chain management is supposed to run on a single network. This network will be the basis of the Hyperledger Fabric network. This network is called the "supplychain-network". When this research was carried out, there was still no cross-channel feature in the Hyperledger Fabric supported by Hyperledger Composer, therefore the network was only designed to run on one channel, namely "mychannel".

The organization is divided into three, where Org1 is specifically for Supplier, Org2 for Manufacturer and Org3 for Consumers, as shown in Fig. 1, and each Orgs have different Admin and all of these admins have access to install any defined chaincode on the peer. One peer can run many chaincodes according to the machine's capabilities and needs.

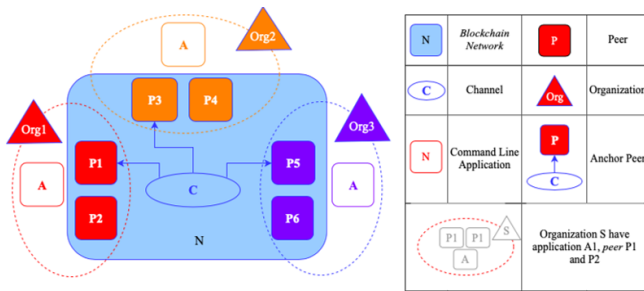


Fig. 1. System block diagram

The supply chain management process begins in parallel, as seen in Fig. 2 because a Supplier can have many Manufacturers, as well as a Manufacturer, can also have

many Suppliers. Suppliers admins can create the Purchase Order (PO) to order commodities from the desired suppliers. The commodity will then be changed in status and will be sent if the Supplier receives a PO from the Manufacturer. Just as with the Supplier, only the Manufacturer admins can create the PO.

At the Manufacturer, the Manufacturer admin will be responsible for carrying out POs against suppliers, move commodity stocks that are deemed necessary to be moved to another Manufacturer supplier, and accept PO requests made by the Consumers. Members can only change commodity status and send commodities that have been received and confirmed by the admin. For the Consumers, the Consumers admins are responsible for creating PO the Manufacturer, as well as to approve the return, installation, and determine whether the commodity is included in the balance or not, while the non-admin is only responsible for changing the commodity status.

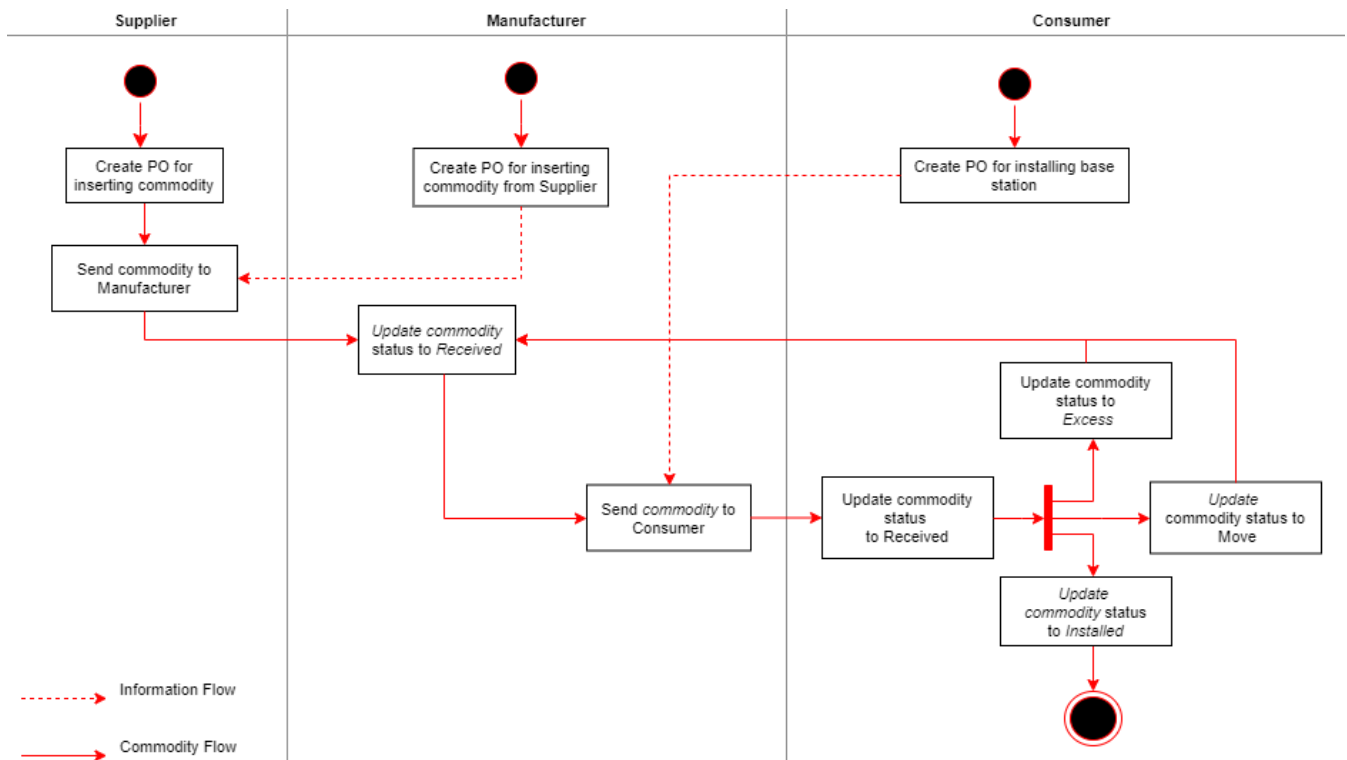


Fig. 2. The system blockchain flow network

IV. RESULT ANALYSIS

In this paper, we conduct two tests, which is a function using query and without query. Because query can only be used on Read function, in the second test, we will only compare the Read function with a query and without a query.

A. Comparison of Create, Read, Update and Delete Duration

The purpose of this test is to find out which function has the fastest duration between Create, Read, Update, and Delete. By comparing the four functions using box plot graphs, it will show how long the duration of the functions are and the deviations that occur in each iteration to access the data directly through the transactions that have been

designed. Comparison is divided into three categories, which are 10 commodities, 100 commodities, and 1000 commodities. It can be seen in Fig. 3 that Read function always has the shortest duration needed and

Update has the longest duration needed.

These results occur in all categories even in category 10 commodities has a narrow quartile distance as in Fig. 2 section a, which indicates that most iteration results do not have a significant deviation, and the distance between quartiles is in a short duration area. The Delete function has a duration that is not much different from Create function, but in category 100 commodities, the duration of the Delete function is much

shorter than the duration of Create function.

Based on Fig. 3, it can be concluded that the duration of the Read function has the highest result stability among other functions because, in all three categories, the Read function has the lowest quartile distance compared to all other

functions, followed by Create and Delete functions. On the other hand, Update is a function that has the lowest stability compared to all other functions. Read function is 33% faster than Create function, 69% faster compared to Delete function, and 82% faster compared to the Update function.

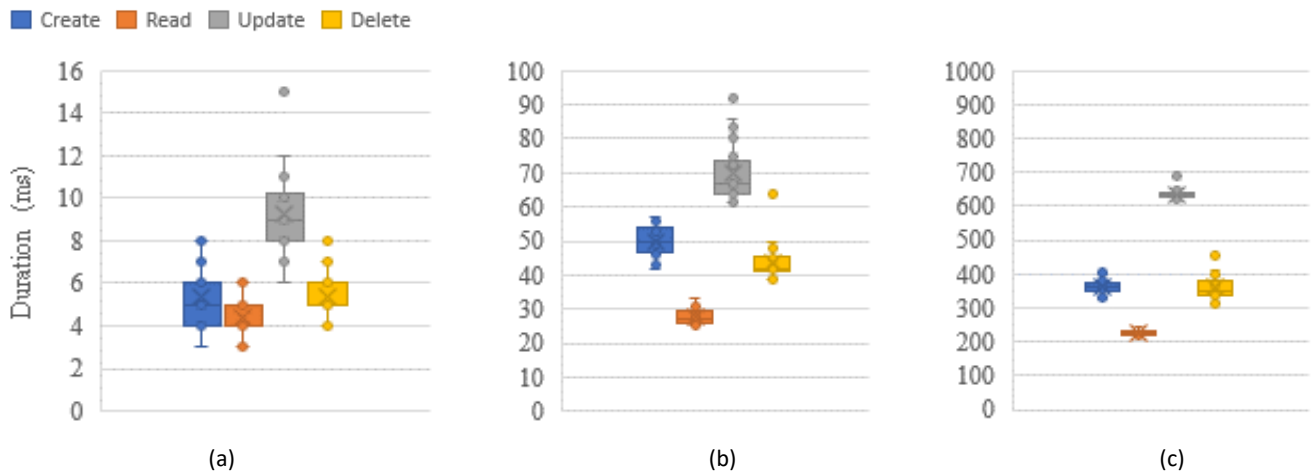


Fig. 3. Average duration comparison of all function for a) 10 commodities b) 100 commodities and c) 1000 commodities

B. Comparison of Read Function with Queries and Without Queries

The purpose of this test is to find out how the influence of query usage in the process uses the Read function. The query used in this test is the simplest query possible in Hyperledger Composer. By comparing the two conditions using box plot graphs, it can be compared how long it takes for the two conditions to access commodities data directly through the transactions that have been designed and the deviations that occur in each iteration. Comparison is divided into three categories, which are 10 commodities, 100 commodities, and 1000 commodities.

Based on Fig. 4, using queries on the Read function has a very significant impact wherein category 10 commodities, the average duration of Read function increases up to 12238.9% or 122 times the duration needed for Read function without using a query. In 100 commodities category,

the increase in average duration is not as significant in category 10 commodities, where the increase occurs as much as 1950% or 20 times the duration of Read function without using a query. In the 1000 commodities category, the increase in average duration is not as significant as 10 commodities category, where the increase occurs as much as 308% or three times the duration needed for Read function without using a query. This causes the efficiency of data retrieval to be lower by using the query, although it is undeniable that the use of queries greatly facilitates retrieval of data on this system. We figured out that that duration of Read function using a query is not directly affected by how long the block is being read based on the duration of 10 commodities and 100 commodities category.

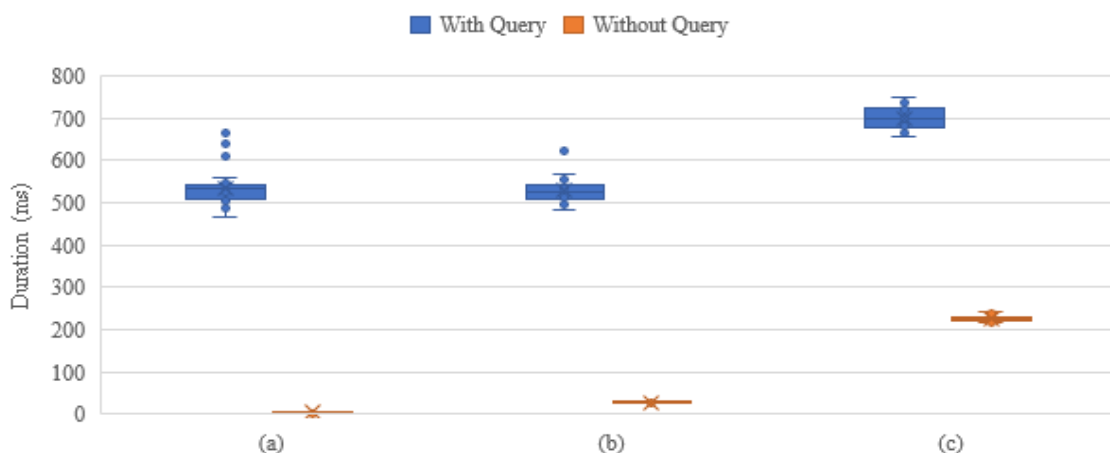


Fig. 4. Average duration comparison of read Function with Query for a) 10 commodities, b) 100 commodities, and c) 1000 commodities

V. CONCLUSION

The innovation of blockchain integration in the supply chain is promising and can improve the supply chain in many factors. In this research, we proposed a blockchain based supply chain by using Hyperledger Fabric and Hyperledger Composer as its blockchain framework. We presented our system and measured the throughput of the system to describe how the blockchain system behaves when processing data with different sizes. The experimental results show that in the blockchain using Hyperledger Fabric and Hyperledger Composer,

Read function has the shortest duration needed, which is 33% faster than Create, 69% compared to Delete, and 82% compared to Update. The update function has the longest duration needed. We also found out that while using query in Read function is slower than without using a query with the increase of the average read duration needed over 10 commodities category by 12238.9%, over 100 commodities category by 1950% and over 1000 commodities category by 308% than reading without using a query, and the read function with a query is not directly affected by how long the block is being read. We hope that our research would help as a guide to design and implement a supply chain management system using blockchain technology.

ACKNOWLEDGMENT

This research is part of the study requirement of the master's degree program at the Electrical Engineering, Faculty of Engineering Universitas Indonesia.

REFERENCES

- 1 S. Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008, doi: 10.1007/s10838-008-9062-0.
- 2 W. T. Weilbach and Y. M. Motara, "Applying distributed ledger technology to digital evidence integrity," *SAIEE Africa Res. J.*, vol. 110, no. June, pp. 77–93, 2019, doi: 10.23919/SAIEE.2019.8732798.
- 3 C. Cachin and M. Vukolić, "Blockchain Consensus Protocols in the Wild," 2017.
- 4 P. Helebrandt, M. Belluš, M. Ries, I. Kotuliak, and V. Khilenko, "Blockchain Adoption for Monitoring and Management of Enterprise Networks," *2018 IEEE 9th Annu. Inf. Technol. Electron. Mob. Commun. Conf. IEMCON 2018*, pp. 1221–1225, 2019, doi: 10.1109/IEMCON.2018.8614960.
- 5 K. Gai, Y. Wu, L. Zhu, L. Xu, and Y. Zhang, "Permissioned Blockchain and Edge Computing Empowered Privacy-Preserving Smart Grid Networks," *IEEE Internet Things J.*, vol. 6, no. 5, pp. 7992–8004, 2019, doi: 10.1109/JIOT.2019.2904303.
- 6 P. Cui, J. Dixon, U. Guin, and D. Dimase, "A Blockchain-Based Framework for Supply Chain Provenance," *IEEE Access*, vol. 7, pp. 157113–157125, 2019, doi: 10.1109/ACCESS.2019.2949951.
- 7 Armayah, Sumardi, K. Damang, and M. Munizu, "Supply chain collaboration and its effect on SMEs' competitiveness of seaweed business sector in Takalar Regency," in *IOP Conference Series: Earth and Environmental Science*, 2019, vol. 235, no. 1, pp. 0–6, doi: 10.1088/1755-1315/235/1/012015.
- 8 T. T. A. Dinh, R. Liu, M. Zhang, G. Chen, B. C. Ooi, and J. Wang, "Untangling Blockchain: A Data Processing View of Blockchain Systems," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 7, pp. 1366–1385, 2018, doi: 10.1109/TKDE.2017.2781227.
- 9 J. Sousa, A. Bessani, and M. Vukolic, "A byzantine Fault-Tolerant ordering service for the hyperledger fabric blockchain platform," *Proc. - 48th Annu. IEEE/IFIP Int. Conf. Dependable Syst. Networks, DSN 2018*, no. 1, pp. 51–58, 2018, doi: 10.1109/DSN.2018.00018.
- 10 T. Lepoint, G. Ciocarlie, and K. Eldefrawy, "BlockCIS - A blockchain-based cyber insurance system," *Proc. - 2018 IEEE Int. Conf. Cloud Eng. IC2E 2018*, pp. 378–384, 2018, doi: 10.1109/IC2E.2018.00072.

- 11 S. Wang, L. Ouyang, Y. Yuan, X. Ni, X. Han, and F. Y. Wang, "Blockchain-Enabled Smart Contracts: Architecture, Applications, and Future Trends," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 49, no. 11, pp. 2266–2277, 2019, doi: 10.1109/TSMC.2019.2895123.
- 12 A. R. Rajput, Q. Li, M. Taleby Ahvanooey, and I. Masood, "EACMS: Emergency Access Control Management System for Personal Health Record Based on Blockchain," *IEEE Access*, vol. 7, pp. 84304–84317, 2019, doi: 10.1109/ACCESS.2019.2917976.

AUTHORS PROFILE



Nur Hazbiy Shaffan, was born in Malang, in 1997. He was graduated in Computer Engineering Undergraduate Degree in Universitas Indonesia (2019) with a thesis in the field of Smart Contract in Private Blockchain. He is currently pursuing a master's degree at Electrical Engineering Universitas Indonesia (2019 until now) with a fast track program, focuses on consensus algorithm, and currently working on blockchain

scalability with clustering blockchain.

The first research was proposed about the blockchain system. His research fields' interests are blockchain, big data, artificial intelligence, and embedded systems. In the blockchain, his interests are consensus algorithm, blockchain security and encryption, and also blockchain consortium.



F. Astha Ekadiyanto, received his bachelor's degree in Electrical Engineering, from Universitas Indonesia with a thesis in the scope of Low Powered Microelectronic Systems in 1995. His master's degree of Science was from Universitaet Duisburg-Essen, Germany (2005), in Computer Science and Communication Engineering with the thesis in the scope of active networking.

Currently, he is pursuing his Ph.D. in the field of Computer Engineering, focusing on Peer-to-peer service provisioning support to computer-supported collaborative work applications at the Institute for Computer Engineering, Universitaet Duisburg-Essen. His research fields extend from distributed computing to cyber-physical systems and currently working on embedded systems and low-level software-defined networking.



Anak Agung Putri Ratna, was born in Japan in 1961. She was graduated from Electrical Engineering Universitas Indonesia in 1986 and continued at Electrical Engineering Waseda University for the Doctoral Program. She was a lecturer in the Department of Electrical Engineering Universitas Indonesia from 1987 until now.

Her research area focuses on computer architecture, computer networking, and the e-learning model, mainly on Applied Sciences in Technology. She is mostly working on the artificial intelligent algorithm in plagiarism detection. In artificial intelligence, the focus is improving efficiency for AI algorithms on various applications, such as Latent Semantic Analysis Automatic in language essay grading system and WinoWing Algorithm in Cloud-Based Implementation and WinoWing Algorithm in Cloud-Based Implementation.