Decentralized Payment Aggregator: Hyperledger Fabric

Md.Al-Amin[1]
Faculty
Department of Computer Science
Faculty of Science And Technology
American International
University-Bangladesh

Khondoker Shahrina[2]
Department of Computer Science
Faculty of Science And Technology
American International
University-Bangladesh

Rubyet Hossain[3]
Department of Computer Science
Faculty of Science And Technology
American International
University-Bangladesh

Debashish Sarker[4]
Department of Computer Science
Faculty of Science And Technology
American International
University-Bangladesh

Sumya Sultana Meem[5]
Department of Computer Science
Faculty of Science And Technology
American International
University-Bangladesh

***Abstract*—Blockchain has become a great trend and very popular in the present era. There are two types of Blockchain technology, centralized and decentralized. In this research, the main concern is about the decentralized payment gateway, which is a trustworthy architecture and does not depend on third parties. For recording the transaction, decentralized payment systems use distributed ledger. Previously, Bitcoin and Ethereum payment systems were used to verify the consistency of the ledger of blockchain and also the transaction data along with the sender-receiver address and transaction value, but as all the payment system is public, so the transaction mode is also public. However, here the main concern is privacy and security. Because anyone can easily access the network, the attacker can also attack the network and the identity and transaction records and the address of the user identity, which is a privacy challenge. This research incorporates the Hyperledger Fabric, which is private, to overcome this challenge. Moreover, no one can access it from outside of the network. The transaction cost is low and the timing is fast during transactions. Considering the above scenario, this research proposes a decentralized payment system architecture using Hyperledger Fabric.**

***Keywords*—Blockchain; decentralized; hyperledger fabric; bitcoin; payment system**

I. Introduction

In the present world, transactions have become one of the communication mediums. So it is imperative to ensure that the transaction should be safe and secure. Because of the many uses of distributed ledger technology, some companies have built their version of Blockchain systems to meet their requirements, resulting in the in-house creation of many blockchain solutions using multiple systems and architectures. Such projects use various technologies and consensus methods tailored to specific applications and use cases. Because of the vast number of initiatives, development has become increasingly fragmented, with no or little compatibility across various blockchain protocols [1]. In the earlier, third-party interference was the only medium for transactions. However, it was proven to be very insecure. The increasing number of online platforms has raised the necessity of payment systems. Payment gateways enabled online platforms to integrate multiple payment methods at once, often leading to more sales and engagement to the online platform. In short, many payment gateways are available, including international and local ones. Currently, they are built on a centralized system and use several strategies to minimize the cost of each transaction. They often tend to save up the money received from customers and hold the money for several days to earn revenue from the saved money. This method causes the merchants not to be able to receive money immediately but reduces the cost of the transferred money. Hyperledger Fabric, a distributed ledger platform, solves the problem of reducing third-party dependencies and removes double spending issues. It is a successful innovation that ensures money is spent only once. Decentralized systems are a platform where they can interact with other parties without involving the third party or any intermediaries. Hyperledger Fabric is a decentralized permissioned blockchain infrastructure and the fastest open source permissioned blockchain [24]. It is challenging to hack because it uses mathematical computation. Chaincode is the primary smart contact in the Hyperledger Fabric. Chaincode is executed by the peer nodes of a Fabric network, accessing the ledger data and endorsing transactions. Mainly the architecture of Hyperledger Fabric provides flexibility and a high degree of resilience in implementation and design. The flexibility of design achieves privacy, scalability, and other essential attributes. Blockchain service is the central part of the Fabric. Hyperledger Fabric chain code allows any programming language but commonly used Go language, JavaScript and Java, but appropriate modules should be installed [2]. This paper aims to construct a decentralized payment gateway that allows unlimited payments of any amount, is more secure, low cost in transactions, gives the best performance, and the main thing that Hyperledger Fabric ensures is the system's security. The problems of a centralized payment gateway can be understood from the above discussion and how decentralized payment gateways can solve them. In the Asian subcontinent, there is no

service to address the issues of a centralized payment gateway. To address this issue, this work presents a gateway to improve the payment service.

A. Research Goal

1) Research Objective: This research aims to construct a decentralized payment system that will solve the loopholes that come with the centralized payment system.

2) Significance of Research: Currently, centralized payment systems are using several strategies to lessen the cost of transactions and earn revenue from pending transactions. A decentralized payment system can eliminate this issue and make the process more secure and efficient. This research is taking place to improve the payment service by introducing a decentralized payment system.

3) Research Question: The main purpose of conducting this research can be expressed as below:

1) What are the problems of centralized payment systems?
2) Will decentralized payment systems eliminate or lessen the existing problems?
3) How will people be benefited from a decentralized payment system?
4) What will be the challenges of introducing a decentralized payment system?

II. Related Works

Decentralized cryptocurrencies like as Bitcoin [3], Hyperledger Fabric [4], Zcash [5] and Ethereum [6] have been a trending issue, and blockchain technology, received increasing attention. Currently, the blockchain plays a significant role also in financial sector [7]. In previous before blockchain invention, many authors proposed different protocols and solutions to make the transaction procedure secure and easy way.Such as,The authors of [8] presented is a payment mechanism for peer-to-peer (P2P) commercial transactions. It facilitates electronic currency transactions between purchasers and vendors. In this arrangement, financial companies become collaborators in their clients' Internet-based e-commerce transactions [23]. The uniqueness of the proposed approach is the restriction of the finance sector' engagement to supplementary services and support but in that circumstance there were some drawbacks regarding the security. Blockchain invention by S.Nakamoto [9] brought also progress in the financial sector where A peer-to-peer formed of electronic currency which would enable internet payments to be transmitted through one party to another, bypassing banking institutions. The main concern of the solution was to remove double-spending issue [10]. All the transactions are hashed into a chain which is hash-based proof-of-work of that network and after establishing record cannot be modified without repeating the proof-of-work. It provides the main security issue. In article [11] the authors describes a distributed payment method which is based in payment tokens that uses Blockchain technology to safeguard consumers from data theft caused by illegal use of their payment card information owing to payment system inefficiency. Furthermore, the privacy and security of the system is consolidated by a permissioned Blockchain consortium. Using blockchain technology, the authors [12] create a centralized payment counter money transfer mechanism. Blockchain technology inherent decentralization, distributed verification, transparency on transmission, as well as other features, the method not only recognizes that transaction records can also increases payment security, and completely protects client privacy. It eliminates the issues of money usage security and danger created by inappropriate teller operation in the standard counter cash withdrawal method. In [13] the authors proposed a method called Fastpay where the authors combined the concepts of blockchain technology and IoT [14]. FastPay, which ensures the security of quick payments in blockchain-powered edge and IoT systems. FastPay is based upon the smart contract concept, which is implemented by several well-known blockchain platforms, including Ethereum [15]. The system ensures quick payments by requiring consumers and guarantors to produce transaction proofs before payment is done. With these evidence, consumers and guarantors will face harsh fines if they attempt double-spending scams [25]. A payment takes 9 seconds to confirm.The authors [16] approach a system that supports late payments using our novel idea known as latent-transactions. This study also incorporates the tools for making effective off-chain contracts between the entities engaged in a transactions. As a result, one may profit from a service before paying for it. Furthermore, the system provides a service expansion mechanism by offering a catalog to which any Blockchain user can register and one may offer services with flexible pricing based on the context such as need, activity at execution time, etc. The authors proposed an algorithm using blockchain concept. The algorithm will enable users to operate on blockchain networks using cryptocurrencies. It differs significantly from the current system in that customers would be able to exchange without the assistance of third parties, and merchants will also be eased with their transaction. This form of transaction will be highly convenient for both buyers and sellers. Consumers and suppliers may both have this whole transaction time, date and all the things they engaged with while the transaction has been taken place [17].

III. Background Study

A. Blockchain

Blockchain service contains P2P Protocol, Distributed Ledger and Consensus Manager, these three major components. P2P Protocol provides capabilities like flow control and bidirectional streaming as it uses Google RPC [18]. It works along with existing internet system [9].

Distributed ledger calculates a crypto hash after each block, transmits minimal delta of changes of out of sync peers and minimize the size of data efficiently to manage the world state and blockchain. It uses a special database environment called RocksDB [19] to preserve the dataset. Other large files are not stored on the ledger but their hashes can be stored on the ledger. There are two types of transaction Hyperledger supports: code-deploying and code-invoking. Code-deploying transactions are to install, update or remove a Chaincode. On the other hand, code-invoking transactions are to execute a Chaincode function through API call. Consensus Manager accepts transactions and decides the way to organize and the time to execute a transaction depending on algorithm. If a transaction is successfully executed, it causes the ledger to be changed. Hyperledger has implementation of PBFT (Practical

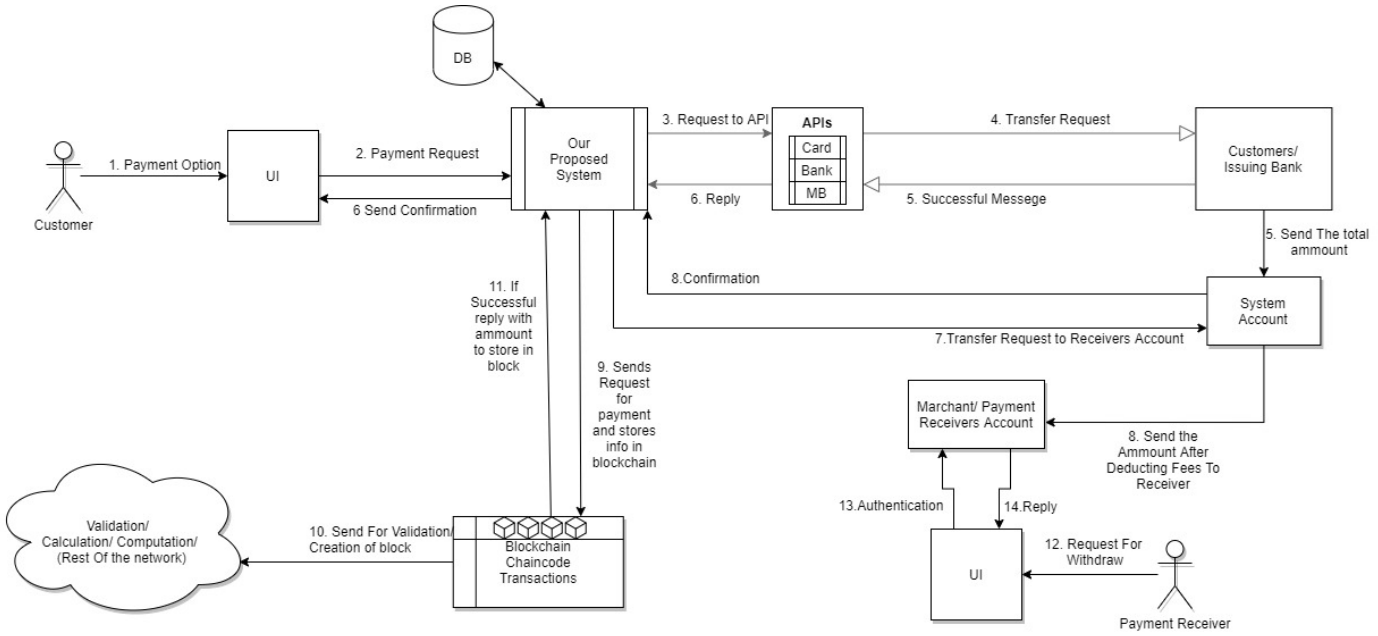

Fig. 1. Architecture of Proposed System.

Byzantine Fault Tolerance) which ensures advanced scalability and fault tolerance.

B. Hyperledger Fabric

Now a days blockchain technology is the most emerging technology and it can be implemented in many different ways. Research is going on this field and researcher create different types of DLT(distributed ledger technology) for different purposes and construction [26]. In present moment three of the most popular DLT's are "Etherium"2 [6], "R3 Corda"3 (Corda) [20], "Hyperledger Fabric"4 (Fabric) [6]. These DLT's have different visions of the application of blockchain in different fields. Etherium is the independent and it is a less and public approval DLT. On the other hand, Hyperlagger Fabric and Corda DLT are allowed. Corda focuses on the financial services industry; Fabric focuses on developing a modular and extensible architecture that can be used in various industries, with a focus on banking and healthcare services, the supply chain.Hyperlenger takes a unique method to the traditional blockchain model, in part by manipulating the admission of participants to its base. In other words, Hyperlagger is an authorized shared ledger. Hyperlegger records computational cycles, reduces scales and meets the multitude of case requirements. Hyperlenger is based on the expectation that there will be many blockchain networks, with each network registration serving a different purpose. Although there is a single popular instance of a common network, it is not necessary for a network registry to rely on another network for its basic functionality. Despite this network independence, Hyperledger still needs an addressing system that allows transactions in one registry to identify and use appropriate transactions and chain code in other registers.

C. Chaincode

Chaincode service hosts the Chaincode using Docker which yields a lightweight and secure system for Chaincode execution. Secured Docker Registry of Hyperledger images and Chaincode containing custom images are enabled by Secure Registry Services. Alike invocation of webservice or database call, Chaincode transactions are configured while deploying the Chaincode and they are time bounded [27]. The ledger will not update if a transaction runs out of time. After reaching consensus for a block, the database is committed with the changes and number of world state block increments. If peers fail to reach consensus, the update is discarded and database no not change.

IV. Proposed System Modules

As discussed earlier, the proposed system designed a payment gateway that will allow Sender to pay directly to the receiver using a decentralized network. The proposed have designed the system in a manner that if any organization or individual wants to receive payment they can integrate our API to their website and Immediately start getting payment. Unlike other centralized payment gateways it doesn't delays the payment. The system will autonomously handle all the account creating process, transactions, building blocks, Committing to the blockchain and retrieving data from the blockchain [29].

A. Architectural Infrastructure

The proposed system has used the linux foundations Hyperledger Fabric [28] as a framework for this project. The whole architecture is designed according to that. Hyperledger Fabrics architecture allows us to create a private network for all the organizations. All the automation part will be handled by the Chaincode. This is the smart contract of Hyperledger Fabric. There will be a central database which will keep the

data while the whole process ends, but to make sure the safety and privacy our system doesn't give access to the database to anyone only accessed by the system itself and also data can't be manipulated because only big data is kept here, and all the information about this is referred to the blocks of blockchain.

Fig. 1 shows how the whole system works from taking the sender/customer sends the money and the receiver receives their payment. The first step is the customer uses the user interface to send money that uses our API to communicate with out automated system. Customer sends the payment request by using their preferred method. Full details about this is discussed in the next modules section.

When the system gets request from the UI then it immediately forwards to the respective bank or card processors. It handles the communication with the bank verifies all the information. If everything is verified and ok then at the same time bank transfers the whole amount to our systems bank and a notification of confirmation to this system.

After the system gets successful transfer notification from the bank, at the same time it sends confirmation to the customer that the payment is complete. At this time our system sends requests to commit this change of asset from customer through different type of payment methods to our bank account. This data flow will be discussed in the net section.

While the first phase is complete it is ready to send the money to the receiver. Receivers will have a fixed account to receive money that he gives while registering. It checks if enough balance is available to be transferred to the receivers account or not. If not, then it waits for some times before transferring. Meanwhile it collects all the money from the all types of payment methods and transfers it to the systems main bank account. Then when there is enough money, Our system makes another request to systems bank to send the rest of amount after deducting the fees from the whole amount to the receivers account.

Bank sends the amount to the receivers account and at the same time gives confirmation to our system. This whole time all the information was kept safe in our system and when these two confirmations are received by our system, it sends the whole request to add them to the blockchain. Then it gets verified and also confirmation is sent back to our system after the transitions is added to the block. This whole verification process is also described later in the modules section.

At the same time there can be more than one transaction from more than one customer. And also, this system is designed to accept all the transactions for all the registered company. For example, two customers from "XYZ" Company is paying online. Both of the transaction will happen individually and as different transaction but it doesn't mean that will create individual blocks. There can be more than one transaction in the block. But using the hashes and timestamp of the block it can easily be found each individual transaction.

Organizations can register with valid information to integrate and use our service and that will also happen using our UI and that information are also saved in to blockchain so that no one can temper those data. The whole operation will run autonomously once the system is up in the network. Here is the Short Pseudocode the of whole operation is provided.

Algorithm 1 Payment Operation for a Customer

```
Output: Transfer money from senders account to receivers account and store the transaction information
m ← Payment Method (Card, Bank, etc.)
r ← The Receiver
a ← The Amount To Be Transacted
if m is card then
    Check Card Validity
    Attempt to perform money transfer to system account
    Set status received from card API
else if m is bank then
    Get redirection URL from bank API
    Redirect to URL with System account number and amount
    Bank API listens to response from the bank
end if
if successful transaction then
    Generate token
    Update Ledger
    Calculate charge from set of rules
    Attempt money transfer (deducting charge) from system account to r
    if successful transaction then
        Update token
        Update Ledger
        Broadcast Ledger
        Return to origin website with status true and token
    else
        Broadcast Ledger
        Return to origin website with status partial
    end if
else
    Return to origin website with status false
end if
```

B. Data Flow:

As in this proposed system have used hyperledger Fabric as our framework while designing the whole architecture we are following the convention and rules of hyperledger Fabric. In our system two transactions will be committed against each payment or money sending.

The first one is the asset transfer from customer or the sender to our bank. In the figure, this first step is represented by Org 1 and Org 2. There are multiple users in Org 1. They represent the payment methods. Our system will get an API response from them when any transfer is done then it will take those information and commit them to the blockchain.

In the 2nd part the system will request a transfer, from our bank to the receivers bank. This asset transfer part is represented using Org 2 and Org 3. Also there are different uses in the Org 3. They represent a company or event. If a company wants to receive the money the endorsement are handled by the p2 and p3 peers. And if events want to receive money then P2 and P4 is the endorsers because there might be different permissions set to different type of peers.

So, the transaction flow for both of the committing steps are like this:

The transaction request is generated from application and

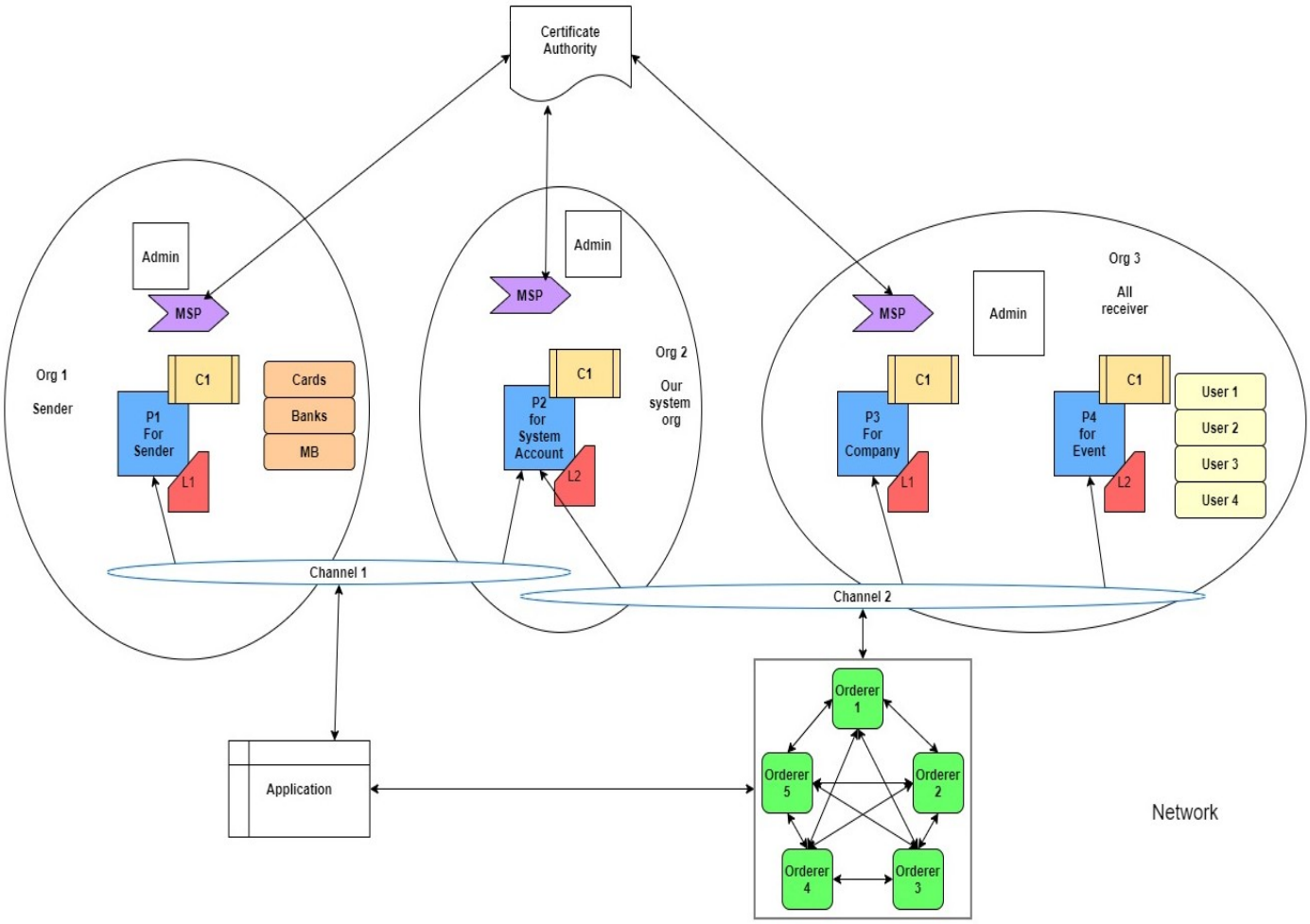

Fig. 2. Data Flow and Committing Procedure into Blockchain.

this application leveraging a supported SDK utilizes one of each Organizations peers to generate transaction proposal. This transaction proposal is send to the peer and these peers are registered through certificate authority. But the responsibilities of the peers are defined by the MSP. That means which transaction is belong to which peer and this peers endorsing and committing permissions are defined by MSP. In each peer there is also a ledger where all the transactions are stored. When the transaction is transfer to the peer, the peer executes the chaincode and validates the transaction but this transaction is not stored in the ledger yet. Only a validate proposal with valid sign is generated and this proposal is verified by the application. If both signed proposal from both peer is matched then it sends this updated proposal to the orderer node.

Meanwhile as there are multiple orderer nodes available, the RAFT consensus protocol is used to choose which orderer will commit the next block. Also, as there are multiple nodes so if one node is down rest of the nodes can build the block.

These orderer node at a time can accept multiple transaction from various application and check the validity of these nodes and generate a block. These blocks are sent to the endorsing peer again to check all those transactions which are endorsed using endorsing peers. If any transactions are remaining in any block which is not endorsed before then this block will consider as invalid block. Otherwise the block will be considered as valid block and add in the ledger of every peer of the network. A pictorial presentation of the scenario is given in Fig. 2.

1) Balance transfer request is generated from the application and it's broadcast through channel 1 to peer1 or p1 which is the peer of org1 and peer2 or p2 which is the peer of org2.
   Note: Every peer is registered or generate an authorized signature through certificate authority and the responsibility of each peer is defined by MSP.
2) This peer actually performs four things 1st one is that the if transaction proposal is well formed or not 2nd one is this proposal is new or has been submitted at past. 3rd one is if the signature is valid. 4th one is that the submitter is fully authorized to perform this specific transaction.
3) This p1 and p2 endorsing peer uses the transaction request as argument to invoke the chaincode.
4) The chaincode is then executed against the current state database to produce transaction results including a response value, read set, and write set but not update

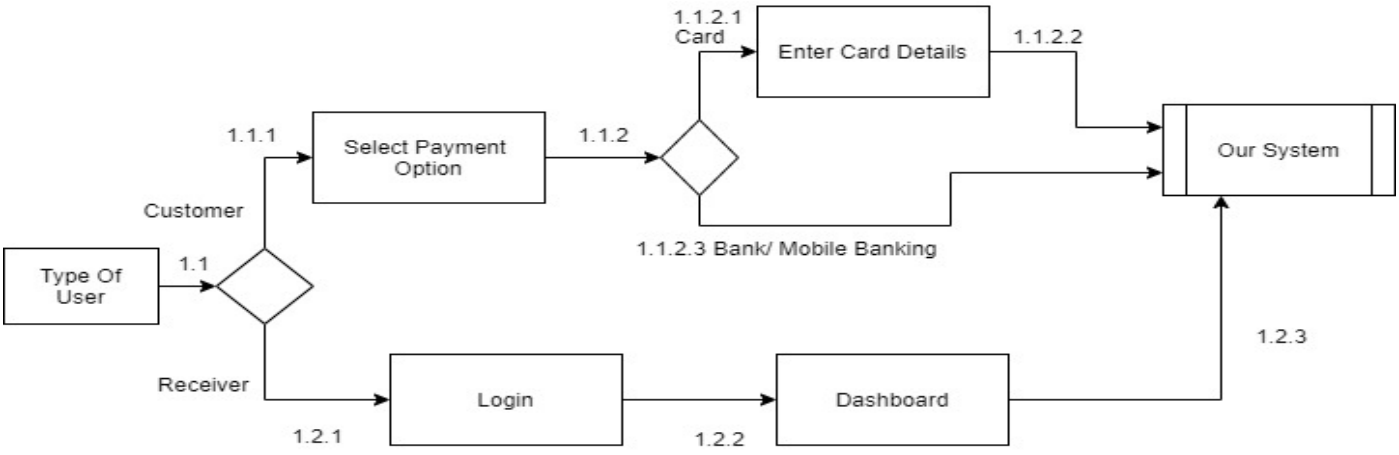

Fig. 3. Data Flow of User Interface Module(Part-1).

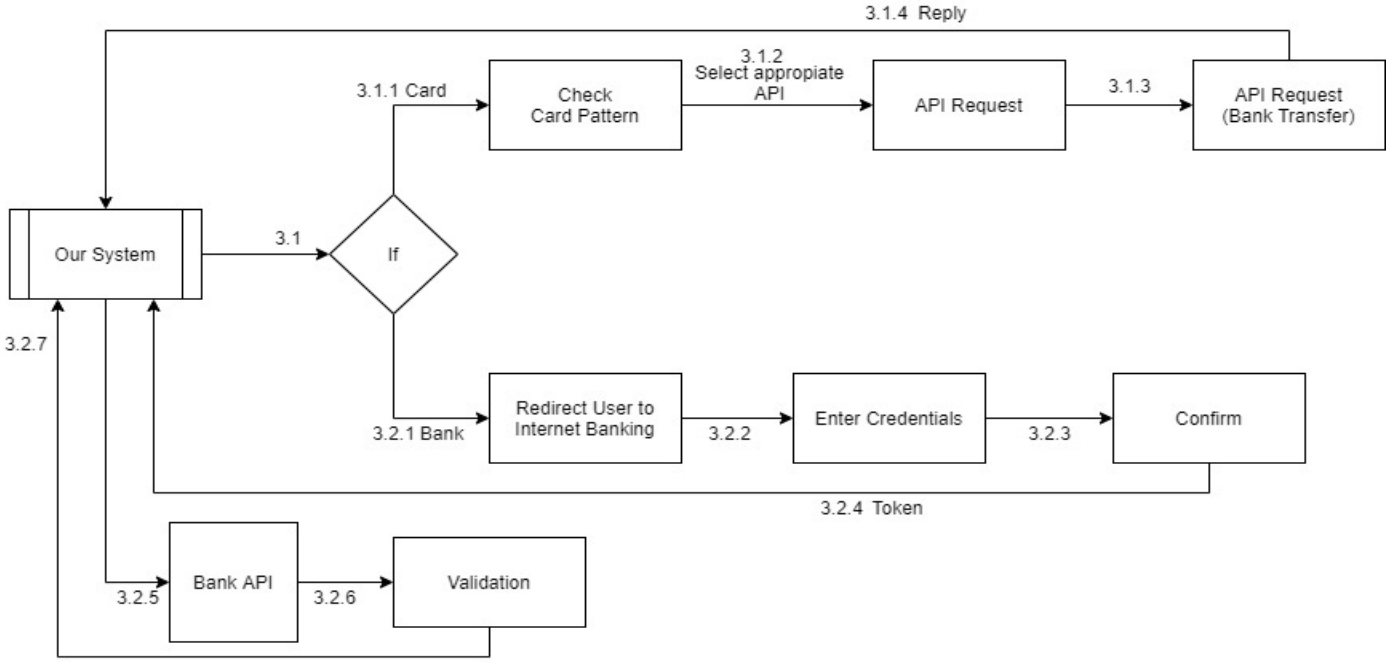

Fig. 4. Data Flow of User Interface Module(Part-2).

the ledger.

5) The set of these values, along with the endorsing peer's signature is passed back as a "proposal response" to the application.
6) The application verifies the endorsing peer signatures and compares the proposal responses to determine if the proposal responses are the same.
7) Then the application sends it to the orderer nodes and in this step the application choose a leader orderer node and remains are working as follower.
8) After receiving the proposal response the orderer node make a block.
9) After creating the block the orderer node then transfer it to all the peers that means p1(org1), p2(org2) and p3, p4(org3) through a global channel.
10) Then these endorsing peers check the signature and the channel ID and if match then commit the block is valid and finally add it in every peers ledger.

C. Modules:

All the operations in the main architecture, We have separated into 3 different modules. These are,

- User Interface Module
- Bank API Module
- Blockchain Module

1) User Interface Module: This module uses our UI to do all the interaction. Customer uses this to select payment method and pay to the receiver. On the other hand, Receiver or the organization can login to their account to see all the

history and other information in their dashboard. The data flow is shown in Fig. 3.

If customer wants to pay, it redirects to the payment option. There are multiple options, for simplicity we are representing bank/mobile baking option and direct card option here. If user selects card option then they put their card details. They can choose to save the details for further use. We use our system to also store the card info to the blockchain if needed. Then the card details is send to system to communicate with the card API. Else if customer choose to pay direct through bank then it requests to out system and it forwards the customer to the internet/mobile banking site for further steps.

From the receiver side, organizations can register to use our service and after successful registration, they can use our interface to login to their account and see their dashboard. In there, all their information is stored and all the histories and all the transactions are displayed.

2) Bank API Module: Whether the customer chooses the card or bank our system requests to the proper API to communicate with the bank. The data flow is shown in Fig. 4.

When the customer chooses the Card option it checks the card pattern and requests to the proper API, Like if pattern is for VISA then it sends request to VISA API, which completes all the processing with the bank and transfers the money to our bank and sends success reply to our system.

If the customer chooses bank, then our system redirects it to the internet banking site or mobile baking site and there they give their credential and follow the steps to transfer money to our system's bank account. When successful, it returns a token. Our system takes that token and verifies with the bank API and if all goes right then the transaction gets completed.

3) Blockchain Module: This module validates and adds data to the block and adds the block to the blockchain [**?**]. As Hyperledger Fabric is used, their system is also being used as the architecture of this module. The data flow is shown in Fig. 5.

1) Initiate to complete the transfer request to Our System (SDK)
2) The SDK sends a transaction proposal to the Peers for the endorsement
3) The proposals are sent to the peers to execute solve and validate.
4) The Peers endorses, performs Chaincode / Smart Contract execution at this point, before endorsing the transaction proposal.
5) After Success it returns to the peers.
6) Then they sign the transaction proposal and returns it to the SDK. (SDk Checks for the validation by comparing to the proposal with the response).
7) If the proposal and Response matches SDK sends request to the Orderer for ordering service.
8) Executes and creates new block.
9) After ordering the transactions, forming block, it broadcasts the block to the Peers
10) The Peers notifies the SDK — the block is committed in the Blockchain for the data also the Peer performs transactions validation and block commitment at this point, before notifying the SDK.
11) The SDK sends a response to the customer and receiver — the block is committed in the Blockchain for the transfer.

This is how the full cycle is done. This gateway uses the decentralized network of Hyperledger Fabric and all the process happens synchronously any autonomously. The system is designed in such a way that no central authority is needed for the operation. And no one can even change any data as there is no control or access over the system. The system only gives read-only data as a query result and only writes data when chaincode is executed, validated and added to the blockchain.

V. Challenges

During the time of research, we faced some challenges. We have established the architecture of the network and its working procedure in this research paper. However, we could not implement the network in a physical device due to time constraints and a lack of proper documentation on the chosen technology. Research should be carried on by implementing the network in a physical device for a better understanding of the network [22]. Due to the nature of the blockchain network, the transactions' validation process takes longer than in a centralized system, leading to a slow completion time for any transaction. Nevertheless, it provides a more secure environment than a centralized system. It is a trade-off between time and security. Blockchain network does not recommend storing large data, as extensive data makes the network vulnerable. As a result, centralized data storage was necessary for extensive data.

VI. Conclusion

In this research, a payment system based on decentralized ledger technology is proposed to keep transactions more secure and allow merchants to receive payment as soon as a transaction is verified. Saving time on receiving the payments and security was the primary concern in this research. We focused on the security and reliability part by using Hyperledger Fabric. A blockchain-based decentralized network gives each user high security and an immutable ledger. Furthermore, private network like Hyperledger Fabric allows reliability by keeping sensitive data private. Current progress indicates the foundation towards building a decentralized payment gateway. This fundamental architecture will help to build a more secure and fast payment gateway and will be able to overcome the shortage of traditional methods.

VII. Future Work

In future, this research can be used to build an actual working gateway and storage system independent from any centralized storage by making the data structure more efficient. Some improvements and more efficient data management can be made in the future. Error handling, such as when the customer successfully sends money to the system account, but our system fails to transfer money to the desired merchant account, is not shown in the research. Future research can work on how the errors should be handled.

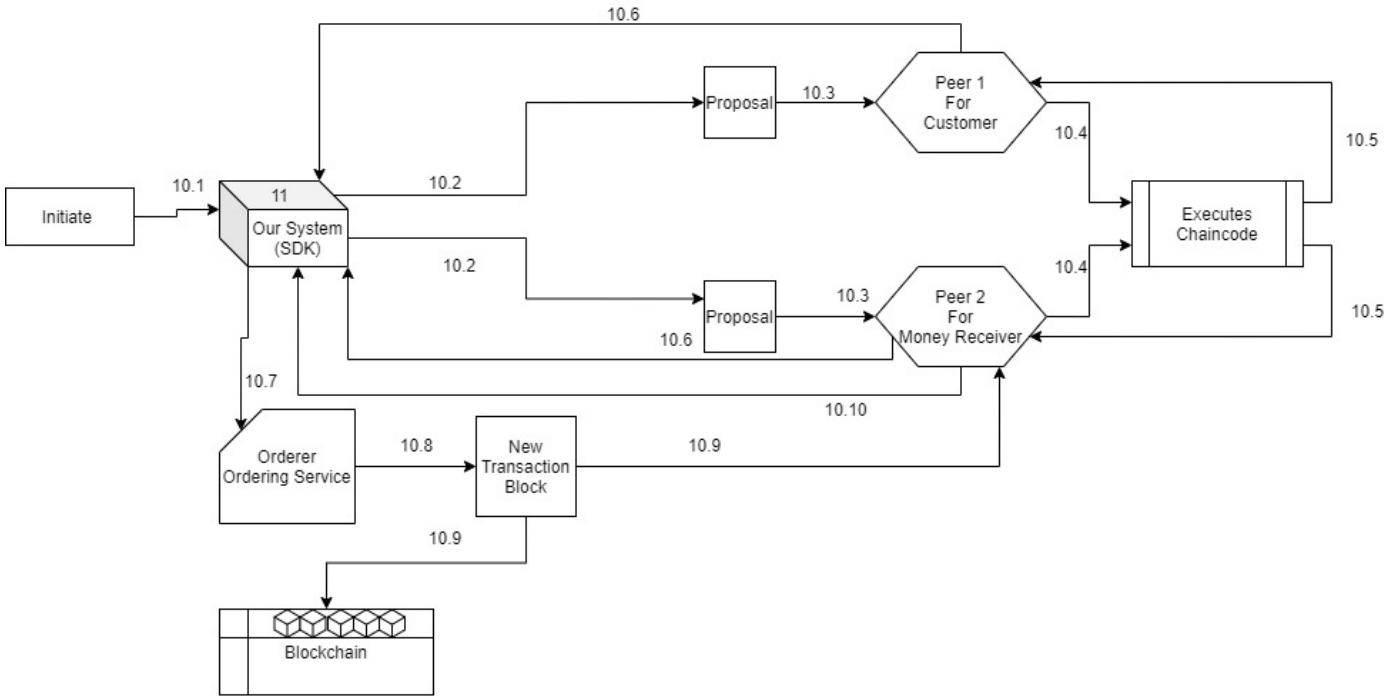

Fig. 5. Data Flow of Blockchain Module.

REFERENCES

[1] Jin, H., Dai, X., & Xiao, J. (2018, July). Towards a novel architecture for enabling interoperability amongst multiple blockchains. In 2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS) (pp. 1203-1211). IEEE.

[2] Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y. and Muralidharan, S., 2018, April. Hyperledger Fabric: a distributed operating system for permissioned blockchains. In Proceedings of the thirteenth EuroSys conference (pp. 1-15).

[3] Nakamoto, S. (2009). Bitcoin: A peer-to-peer electronic cash system Bitcoin: A Peer-to-Peer Electronic Cash System. Bitcoin. org. Disponible en https://bitcoin. org/en/bitcoin-paper.

[4] Aggarwal, S., & Kumar, N. (2021). Hyperledger. In Advances in Computers (Vol. 121, pp. 323-343). Elsevier.

[5] Hopwood, D., Bowe, S., Hornby, T., & Wilcox, N. (2016). Zcash protocol specification. GitHub: San Francisco, CA, USA, 1.

[6] Vujičić, D., Jagodić, D., & Randic, S. (2018, March). Blockchain technology, bitcoin, and Ethereum: A brief overview. In 2018 17th international symposium infoteh-jahorina (infoteh) (pp. 1-6). IEEE.

[7] Cermeño, J. S. (2016). Blockchain in financial services: Regulatory landscape and future challenges for its commercial application. BBVA Research Paper, 16, 20.

[8] Palaka, D., Daras, P., Petridis, K., & Strintzis, M. G. (2004). A Novel Peer-to-Peer Payment System. ICETE (1), 245-250.

[9] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system. Decentralized Business Review, 21260.

[10] Chohan, U. W. (2021). The double spending problem and cryptocurrencies. Available at SSRN 3090174.

[11] Zouina, M., & Outtai, B. (2019, April). Towards a distributed token based payment system using blockchain technology. In 2019 International Conference on Advanced Communication Technologies and Networking (CommNet) (pp. 1-10). IEEE.

[12] Shen, B., & Wang, X. (2021, December). Design of treasury centralized payment counter cash withdrawal system based on blockchain Technology. In ICMLCA 2021; 2nd International Conference on Machine Learning and Computer Application (pp. 1-5). VDE.

[13] Hao, Z., Ji, R., & Li, Q. (2018, October). FastPay: a secure fast payment method for edge-IoT platforms using blockchain. In 2018 IEEE/ACM Symposium on Edge Computing (SEC) (pp. 410-415). IEEE.

[14] Madakam, S., Lake, V., Lake, V., & Lake, V. (2015). Internet of Things (IoT): A literature review. Journal of Computer and Communications, 3(05), 164.

[15] Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. Ethereum project yellow paper, 151(2014), 1-32.

[16] Popa, A. B., Stan, I. M., & Rughiniş, R. (2018, September). Instant payment and latent transactions on the Ethereum Blockchain. In 2018 17th RoEduNet Conference: Networking in Education and Research (RoEduNet) (pp. 1-4). IEEE.

[17] Ahamed, S., Siddika, M., Islam, S., Anika, S., Anjum, A., & Biswas, M. (2021). Bps: Blockchain based decentralized secure and versatile light payment system. Asian Journal of Research in Computer Science, 12-20.

[18] Ahamed, S., Siddika, M., Islam, S., Anika, S., Anjum, A., & Biswas, M. (2021). Bps: Blockchain based decentralized secure and versatile light payment system. Asian Journal of Research in Computer Science, 12-20.

[19] Cao, Z., Dong, S., Vemuri, S., & Du, D. H. (2020). Characterizing, Modeling, and Benchmarking RocksDBKey-Value Workloads at Facebook. In 18th USENIX Conference on File and Storage Technologies (FAST 20) (pp. 209-223).

[20] Mohanty, D. (2019). R3 Corda for Architects and Developers: With Case Studies in Finance, Insurance, Healthcare, Travel, Telecom, and Agriculture. Apress.

[21] Transaction flow." https://hyperledger-fabric.readthedocs.io/en/ release-2.0/txflow.html.

[22] Rajawat, Anand Singh, et al. "Blockchain-based model for expanding IoT device data security." Advances in Applications of Data-Driven Computing. Springer, Singapore, 2021. 61-71.

[23] Hong, Weiyin, and Kevin Zhu. "Migrating to internet-based e-commerce: Factors affecting e-commerce adoption and migration at the firm level." Information & management 43.2 (2006): 204-221.

[24] Cachin, Christian. "Architecture of the hyperledger blockchain fabric." Workshop on distributed cryptocurrencies and consensus ledgers. Vol. 310. No. 4. 2016.

[25] Chohan, Usman W. "The double spending problem and cryptocurrencies." Available at SSRN 3090174 (2021).

[26] Sunyaev, Ali. "Distributed ledger technology." Internet Computing. Springer, Cham, 2020. 265-299.

[27] Foschini, Luca, et al. "HyperLedger fabric blockchain: chaincode performance analysis." ICC 2020-2020 IEEE International Conference on Communications (ICC). IEEE, 2020.

[28] Dhillon, Vikram, David Metcalf, and Max Hooper. "The hyperledger project." Blockchain enabled applications. Apress, Berkeley, CA, 2017. 139-149.

[29] Elghaish, Faris, Sepehr Abrishami, and M. Reza Hosseini. "Integrated project delivery with blockchain: An automated financial system." Automation in construction 114 (2020): 103182.