**UNIVERSITY OF PETROLEUM & ENERGY STUDIES**

**School of Computer Science**

**Dehradun**

**Session Aug – Dec 2020**

**End – Term Report**

on

**Implementation of Blockchain Network**

**(Backend process of storing and managing data in Blockchain)**

**Submitted by**

Adarsh Shukla

### B. tech CSE-CSF (B1) 3rdyear

#### (Enroll No. Rl34218010 & Sap id 500068243)

**Under guidance of**

Mr. Saurabh Jain

(Department of Systemics)

**CANDIDATE’S DECLARATION**

We hereby certify that the project work entitled **“Implementation of Blockchain Network”** in partial fulfilment of the requirements for the award of the Degree of BACHELOR OF TECHNOLOGY in COMPUTER SCIENCE AND ENGINEERING with specialization in Cyber Security and Forensics and submitted to the School of Computer Science, University of Petroleum & Energy Studies, Dehradun, is an authentic record of our work carried out during a period from **August, 2020 to December, 2020** under the supervision of **Mr Saurabh Jain.**

**Adarsh Shukla**

**B.tech CSE-CSF (2018-22)**

**500068243**

This is to certify that the above statements made by me is correct according to the best of my knowledge.

**Date: 25 Nov 2020**

**Mr Saurabh Jain**

Project Guide

**UNIVERSITY OF PETROLEUM & ENERGY STUDIES**

**School of Computer Science , Dehradun**

**ACKNOWLEDGEMENT**

I wish to express our deep gratitude to our guide **Mr Saurabh Jain**, for all advice, encouragement and constant support he has given us throughout our project work. This work can’t be possible without his great support and valuable guidance.

I sincerely thank to our respected Program Head of the Department, **Dr Neelu J. Ahuja** , for his great support in doing our project.

I am also very grateful to **Dr. Manish Prateek (**Dean-SoCS, UPES) for giving us the necessary facilities to carry out our project work successfully.

I would like to thank all our **friends** for their help and constructive criticism during our project work. Finally, I have not any words to express our gratitude to our **parents** who have shown me this world and for each and every support they have given us.

**Name: Adarsh Shukla**

**B.tech CSE – CSF ( 2018-22 )**

**UNIVERSITY OF PETROLEUM & ENERGY STUDIES**

**School of Computer Science , Dehradun**

**Project End – Term Report**

**Project Title**

Implementation of Blockchain Network

**Abstract**

The project aims at taking a step into the field of Blockchain Technology and Cyber Security by developing a blockchain model which can be used for various purposes (like healthcase and education etc.). Blockchain being a decentralized ledger is scalablw as well benefit business through greater transparency. It will be helpful in performing all large computations in a shorter execution time span with efficient speed for scenarios such as healthcare or educational organizations. Using file file handling concept for security in my blockchain network. Whole implementation of this network in C programming only. I used various famous concepts or C programming like, structure, pointer, linked list, file handling, looping and jumping statements etc. Main aim of this project is to show the internal or we can say backend process of blockchain. This project shows how the data is stored and managed in blockchain and the basic role of hashing in Blockchain Technology.

**Keywords**

* Blockchain
* Security
* Hashing
* Healthcare
* Technology
* Decentralization

**Table of content**

**S.No. Contents Page No**

1. **Introduction** 
   1. Introduction to Blockchain 6
   2. Basic Elements of Blockchain 8
   3. Working of Blockchain 9
   4. Hashing in Blockchain 10
   5. Types of Blockchain 11
2. **Problem Statement** 13
3. **Literature Review** 14
4. **Objectives** 16
5. **Methodology** 17
6. **Requirement Analysis** 18
7. **Design**

2.1 Data Flow diagram (Flow – Chart) 20

2.2 Pert Chart 21

1. **Algorithm** 22
2. **Output Screenshots** 25
3. **Conclusion and Future Enhancements** 28
4. **Key-point summery of project** 29
5. **References** 30

**INTRODUCTION**

**Introduction to Blockchain**

In 2008, a groundbreaking paper entitled Bitcoin: A Peer-to-Peer Electronic Cash System was written on the topic of peer-to-peer electronic cash under the pseudonym Satoshi Nakamoto. It introduced the term chain of blocks. After the coming of Bitcoin in industry in 2009, Satoshi Nakamoto remained active in the Bitcoin developer community until 2011. Since then, there has been no communication from him whatsoever, and his existence and identity are shrouded in mystery. The term named as “chain of blocks” invented over the years into the word of blockchain.

Blockchain is the technology that work behind digital currency in which all information about transactions is stored in a digital ledger. Every record is stored in chronological order. All Blockchain records are encrypted, so the information here makes it impossible to hack or attack

Blockchain, which is majorly used for bitcoin, is an emerging technology and is receiving attention these days. The Blockchain serves an unchangeable ledger, providing decentralized transactions between peers. Now a days Blockchain-based applications are rising which are covering many fields like finance, banking sector, healthcare, supply chain etc.

**Technical Definition and terms of Blockchain**

Blockchain is a technology that is peer-to-peer, distributed ledger that will be cryptographically-secure, immutable and updateable only via consensus algorithm or any agreement among peers.

1. **Distributed Ledger**

It is an immutable distributed ledger. It means that a ledger that is spread across the whole network among all nodes/peers in the network and each peer will keep a replica of the complete ledger.

1. **Immutable**

Blockchains are designed to be immutable. Once a block will be written to the blockchain technology, realistically, it cannot change. This provides benefits for audit. As a provider of data you can prove that your data can’t bealtered and for a recipient of data anyone can be sure that the data hasn’t been altered. These benefits will be useful for databases of any financial transactions.

1. **Peer-to-peer network**

In the Blockchain network, there is no central controller in the network, and all peers can communicate to each other directly without any central server or third party involvement such as bank. All the peer node are connected on a simple topology without a hierarchy of central authority or main server, this property making the peer-to-peer network purely decentralized.

1. **Append Only**

One more property in blockchain that we encounter is append-only, it means that data can only be added to the distributed ledger in time-ordered sequential order. This property tells that once data is added to the blockchain ledger, it is not possible to change that data and can be considered immutable. But this can be changed in only rare scenarios where in collusion against the blockchain network succeeds in gaining greater than 51 percent of the power.

1. **Consensus**

In a peer-to-peer network, a consensus mechanism is used to ensure that the block is valid before it is recorded on the ledger. There is a very critical attribute of any blockchain is updateable only by the consensus. This is what provides it the power of decentralization. In this mechanism, no central authority is in control of updating the ledger. Instead, if user want to perform any updation on the blockchain will be against the blockchain validation strict criteria which is defined by the blockchain protocol and added to the blockchain only after a consensus has been reached among all participating nodes or we can say peers on the network. For achieving consensus, we have many consensus facilitation algorithms which ensure that all parties are in agreement with respect to the final state of the data on the blockchain network and agree upon it to be true.

1. **Cryptographically- Secure**

Distributed ledger in blockchain technology is fully cryptographically-secure, it means that cryptography techniques will be used for providing security services which make these ledgers secure against the tampering and misuse. These services include non-repudiation property, integrity of data and data origin authentication. Two main cryptographic primitives are Digital Signatures and Hash Functions.

**Basic elements of a Blockchain**

Basic elements of a simple blockchain will be explained here. These elements are come across in the relation with blockchain:

1. **Block**

A block is merely a selection or we can say collection of transactions bundled together which is organized logically. We can define a transaction is just a record of particular event as the event of transferring cash from a sender's account to the reciever’s account. A block can be made up of transactions and its size can varies depending on the type and design of particular blockchain that in use.

1. **Genesis Block**

A genesis block can be defined as the first block created in the blockchain that is hardcoded during the blockchain was first started. The structure of a block can varies also dependent on the type and design of a blockchain.

1. **Nonce**

A term named nonce in Blockchain is a number that will generated and used only once. A nonce can be used extensively in various cryptographic operations to provide replay protection, authentication and encryption.

1. **Address**

Addresses in the blockchain technology are the unique identifiers used in a transaction to denote the senders and the recipients. An address can be a public key or this can derived from a public key. This will be a good practice for users to generate new address for every transaction in order to avoid linking transactions to the common owner, thus preventing identification.

1. **Transaction**

A transaction in Blockchain Technology is the very basic fundamental unit. A transaction will represent a transfer of any kind of value from one address to other address.

**Working of Blockchain**

This portion explained here a general or basic idea of, how blocks are generated and what the relationship will be between transactions and blocks.

1. A node will start a transaction by creating and then digitally signing it with its private key. A transaction represents various actions in a generic blockchain. It is most common that this is our data structure to represents transfer of any kind of value between users present on the Blockchain network. Transaction data structure can contains of some logic of transfer of value, relevant rules, source and destination addresses and other information as validation.
2. A transaction is propagated (flooded) by using a flooding protocol, called Gossip protocol, to peers that validate the transaction based on preset criteria. Mostly, we required more than one node to verify the transaction.
3. Once the transaction is validated, it is included in a block, which is then propagated onto the network. At this point, the transaction will be considered as confirmed.
4. The newly-created block now becomes part of the ledger, and the next block links itself cryptographically back to this block. This link is a hash pointer. At this point, the transaction will get its second confirmation and the block will get its first confirmation.
5. Transactions are then reconfirmed every time a new block is created. Basically, we required six confirmations in the Bitcoin network or generic blockchain network are required to consider that particular transaction is final.

**Hash Function**

Hash function is mathematical function, which takes any length of data and convert it into a fixed length of data. Input may be of any length like one bit or may be big data but output will be a fixed length size. The values that will return by a hash function will called hash values, hash codes, digests or in very simple terms hash. The values will be used to index a fixed-size table called hash table.

We required 2 security requirements: one is wayness and collision-resistance which are usually required for hash functions. The former ensures that the underlying hash function is not invertible, while the latter implies that it is not easy to find two inputs having the same hash value. For a hash function with n-bit length output, the complexities of breaking one-wayness and finding a collision are respectively bounded by O(2n) brute force attack and O(2n/2) birthday attack.

The very popular and fruitful hash function that can used in blockchains is SHA-256, this is one of the algorithms from the family of cryptographic hash functions named as SHA (Secure Hash Algorithms). SHA is a U.S. Keccak algorithm is the winner in the NIST hash function competition launched by the NIST in year of 2008 and ended in 2012. For satisfying the present time security requirement, SHA-3 and SHA-2 are recommended algorithm for using in Blockchains and Cryptocurrencies.

**Role of Hashing in Blockchain**

The hash function in cryptography field is an integral and most required part of the blockchain innovation. It is most important feature that will gives security capabilities to processed transactions for making them immutable. Hashing will also be at the center of merkle-trees which can be an advanced approach for Blockchain hashing. Blockchain can be very useful in issues of scalability, and mobile/light wallets. Some popular role of hashing in Blockchain are:

* Addresses on the Blockchain are derived from hashing e.g. Bitcoin addresses use SHA2-256.
* Hashing helps in defining cryptographic signatures that help determine valid transactions.
* The hash of a transaction makes it easy to keep track of transactions on the blockchain.
* Hashing functions are crucial in crypto mining where a valid nonce is discovered by computing several hashes.

• Hashing helps to form a consensus on the blockchain.

* It makes the permanent data storage less bulky or simply more economical.
* Hashrate determining how fast and smoothly-running the mining process is.

**Types of Blockchain**

In many books, there are primarily only two types of blockchains: Private blockchain and Public blockchain. But we have several variations too, like Consortium blockchain, Hybrid blockchains etc.

1. **Public Blockchain**

A public type blockchain is a blockchain that will non-restrictive, permission-less distributed ledger system. Any person who has a stable internet connection can sign in on a blockchain platform for becoming an authorized node and can be a part of the blockchain network. A blockchain node of distributed ledger which is a part of the public type of blockchain is authorized to access current and past records, verify transactions and perform proof-of-work algorithm for an incoming block and perform mining. Usually, the basic work of public blockchains is to mine and exchange cryptocurrencies. So, the most common public blockchains are Bitcoin blockchain and Litecoin blockchains. Public blockchains wll be mostly secure if the users strictly follow security rules, regulations and methods. Example: Bitcoin, Ethereum, Litecoin.

1. **Private Blockchain**

A private type of blockchain is a blockchain that will restrictive or permission blockchain operative only in the closed network. Mostly, private blockchains can be used within an organization or enterprises where only selected members are participants of a blockchain network, the level of security, accessibility permissions, authorizations are in the hands of the controlling organization. So we can say, private blockchain is similar in use just as a public blockchain containing a small and restrictive network. Private blockchain networks are used for voting, supply chain management, digital identity, asset ownership mostly.

Examples: Multichain and Hyperledger projects (Fabric, Sawtooth), Corda, etc.

1. **Consortium Blockchain**

A consortium type of blockchain is blockchain that will semi-decentralized type where more than one organization can manage a single blockchain network. This is a difference with it to what saw in a private blockchain, which is managed by only a single organization. More than one organization will act as a user/node in the blockchain and exchange information or perform mining. This blockchains can be used in banks, government organizations etc.

Examples: Energy Web Foundation, R3 etc.

1. **Hybrid Blockchain**

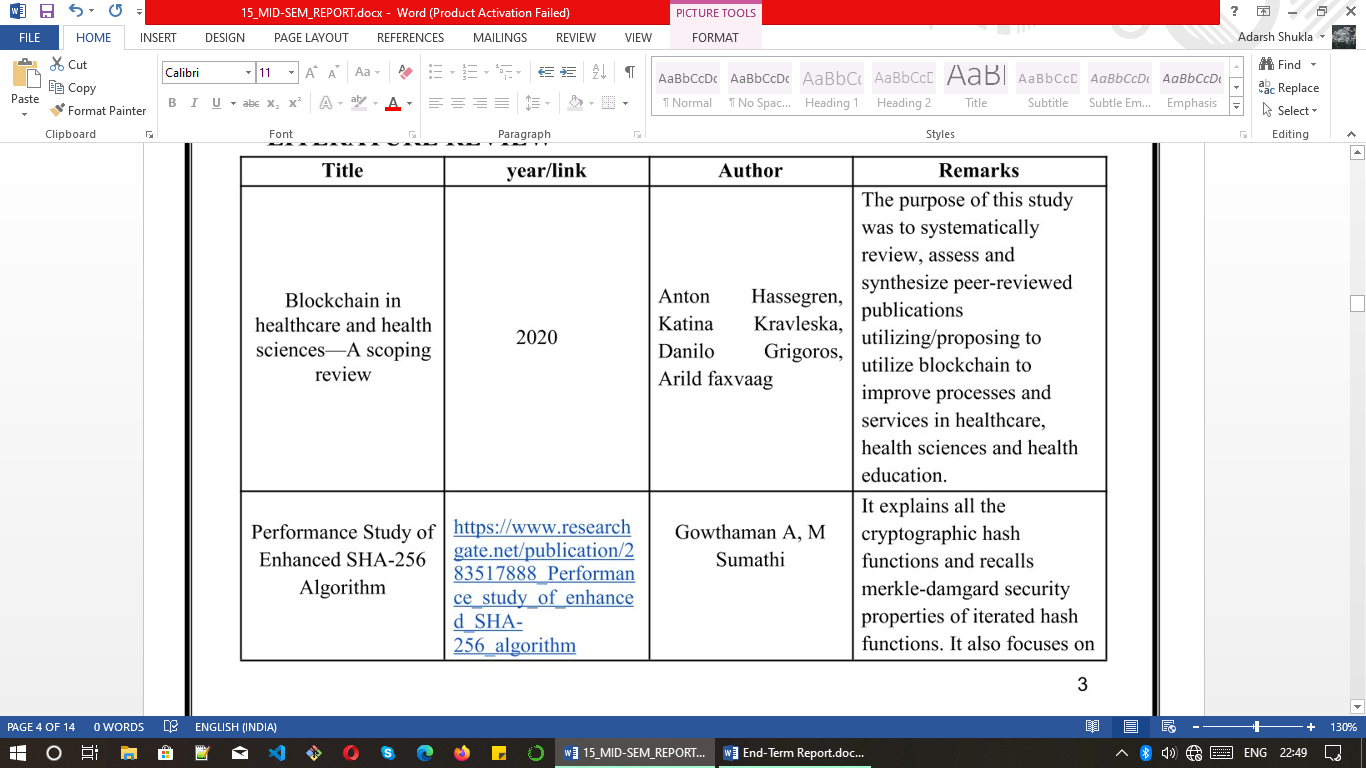
A hybrid type of blockchain is blockchain that will a combination of the private and public blockchain. It can use the features of both (public & private) types of blockchains. This is the one who can have private permission-based system as well as public or permission-less system. With a hybrid network, any user can control who gets access to which data stored in the Blockchain. Only selected section of data or records from the Blockchain will be allowed to go public, rest keeping as confidential like the private network. A transaction within the private network of a hybrid Blockchain is mostly verified in that network but user can also release it in the public section of Blockchain to get verified.

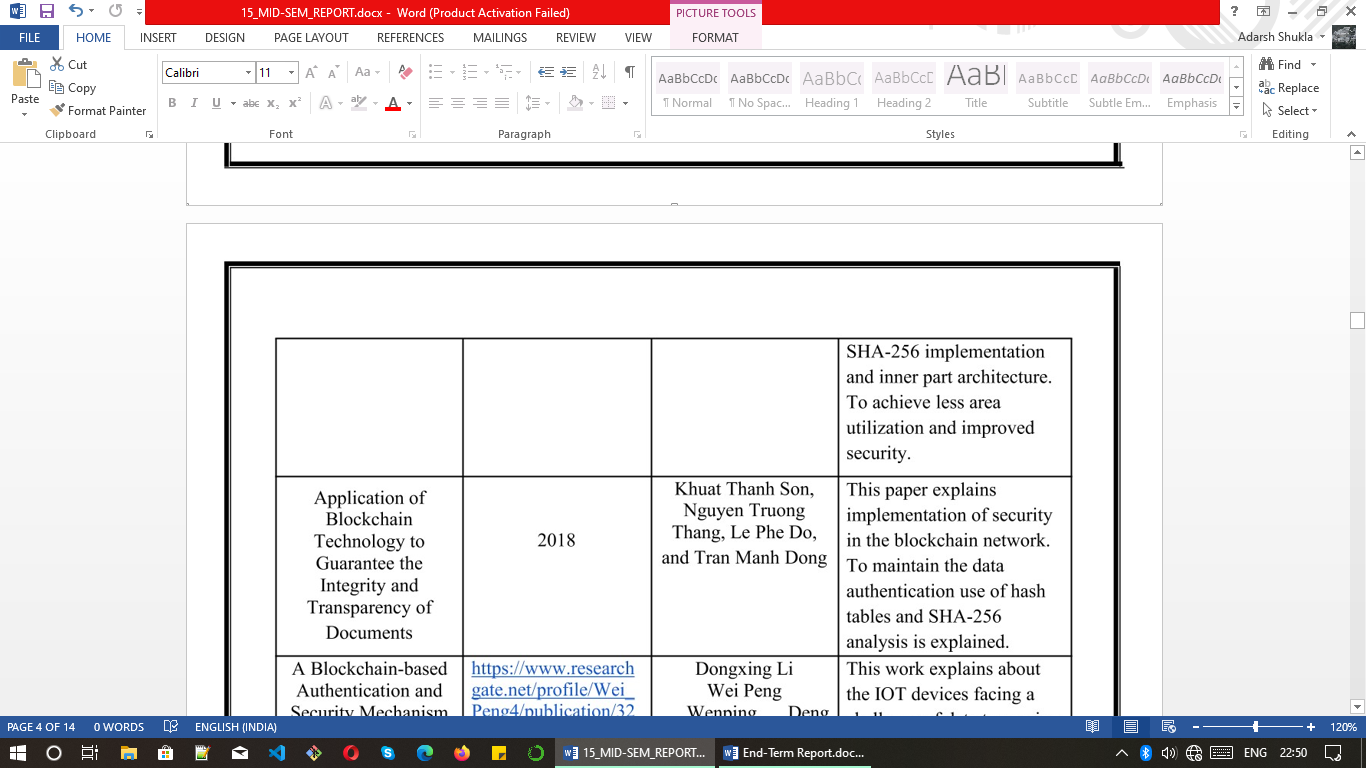
Example: Dragon chain etc.

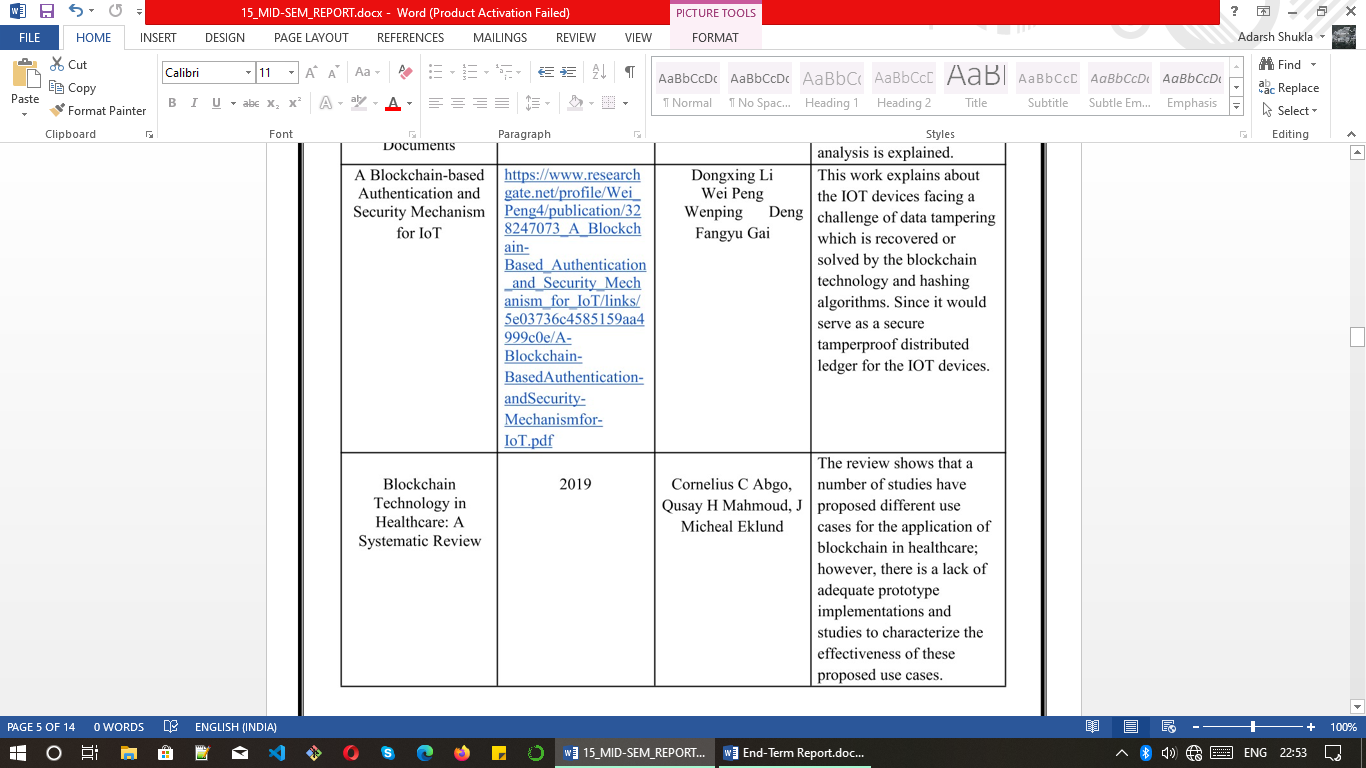
**PROBLEM STATEMENT**

* The upcoming time is based on Blockchain Technology. It will be used in several places (as in for the Healthcare, Education and IOT devices etc.). This new technology has been suggested to disrupt a wide range of data-driven domains, including the health domain. Blockchain has many healthcare use cases including the management of electronic medical records, drugs and pharmaceutical supply chiain management, biomedical research and education, remote patient monitoring, health sata analytics, among others. We have some issues like access control, interoperability, provenance and data integrity that can meant to be improved by Blockchain. Another key characteristic of blockchain is persistency. It is practically impossible to delete or alter entries after being accepted onto the blockchain due to distributed ledger, stored across multiple nodes. Hence maintaining the mobility for records and data in an organization. Use case for our model is a healthcare organization or system. Most of the co-relations in the project would be done with respect to this system in our model. Therefore our primary objective is to develop a simple blockchain model in C language using linked list, structure, pointer, looping and jumping statements. Analyzing and comparing many parameters in the blocks. We will try the concept of file handling as well in C programming.
* One more problem that I experienced personally that blockchain is a new technology but many technologists are not familier with internal or backend process of blockchain distributed ledger. So my project implementation using linked list in C programming is showing the basic understanding of Blockchain distributed ledger functionality.

**LITERATURE REVIEW**





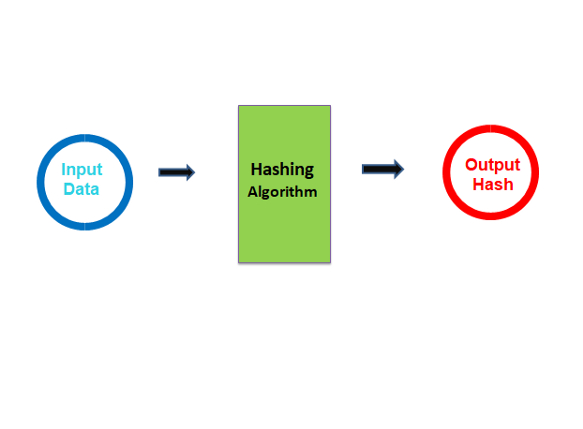


**OBJECTIVE**

The basic objective is to develop a blockchain model and apply SHA-256 hashing algorithms. The input data for the blocks will be entered trough user. File handling concept is a very good option for input due to which our file contents, irrespective of the quality and quantity of the content will be entered in the block making it more secure. Elaborate the backend process of storing and managing data in blockchain distributed ledger is also is a objective of my project.

**Sub Objectives**

* Develop a blockchain model/network at any coding platform in any programming language. I used C language in my project.
* Understand the theoretical concept and role of hanshing algorithm in blockchain technology
* Maintaining the feature of the blockchain network
* Understand the backend functionality of blockchain distributed ledger.

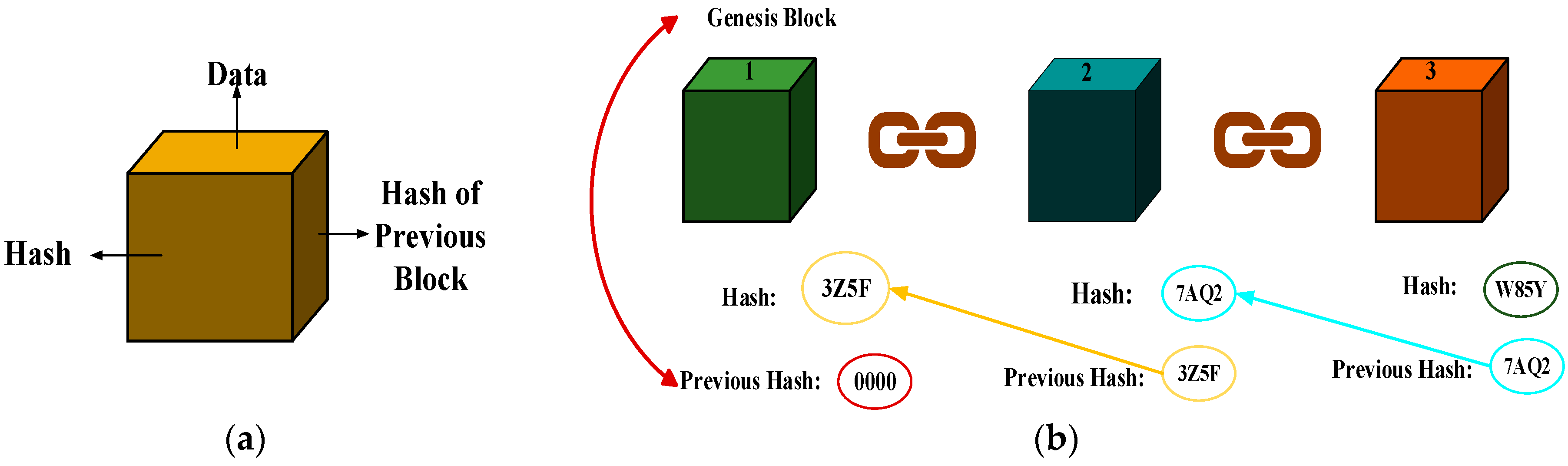


**METHODOLOGY**

For this project, major focus is going to be on the development model in C programming. Use SHA-256 algorithm to calculate hash. SHA-256 function is present in openssl/Crypto.h header file in C language. We will use this library for the hashing mechanism of the blockchain.

The entire implementation of this project can be explained in the following steps :

* The development of the basic blockchain model will be proceeded on any coding platform.
* Number of blocks in blockchin will be entered via user.
* In my model, block’s data will be introduced using file handling. The file type can be different with respect to the user such as text file, excel file etc. Therefore data of file will be stored in the block.
* Applying SHA-256 hashing algorithms (via openssl/crypto.h) to generate hash values.
* The block is mined and is added with the other blocks in our existing blockchain model.
* Every block contains the hash value of the previous block.



**REQUIREMENT ANALYSIS**

In requirement analysis, there are 2 factors: required hardware and required software and resources. Hardware is also most important factor. There is no any meaning of any powerful software with hardware. I have following requirements in my project.

**Hardware :**

* Laptop with i3/i5/i7 processor with 8 GB RAM, 1 TB Hard-disk.

**Software :**

* gcc compiler for C programming/coding platform or IDE

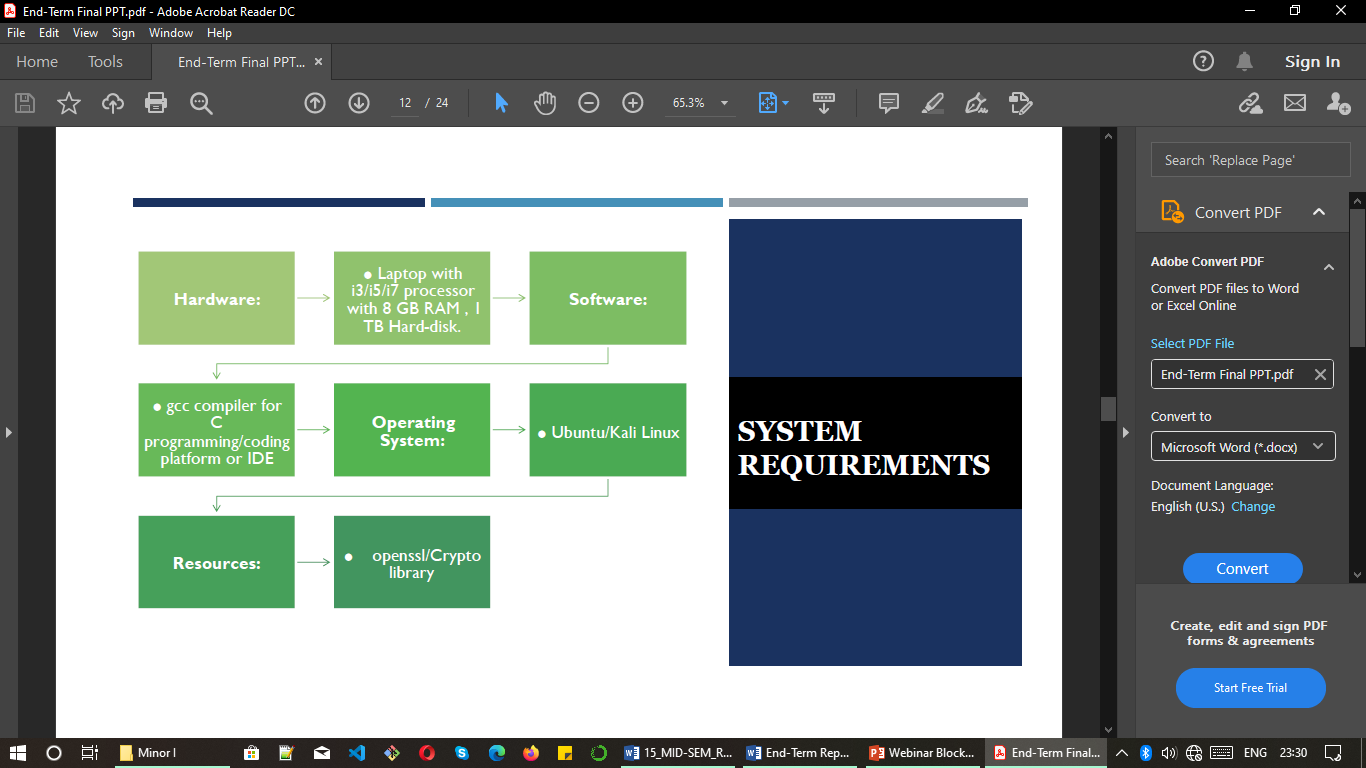
**Operating System :**

* Windows 10

**Resources :**

* Openssl/Crypto.h library

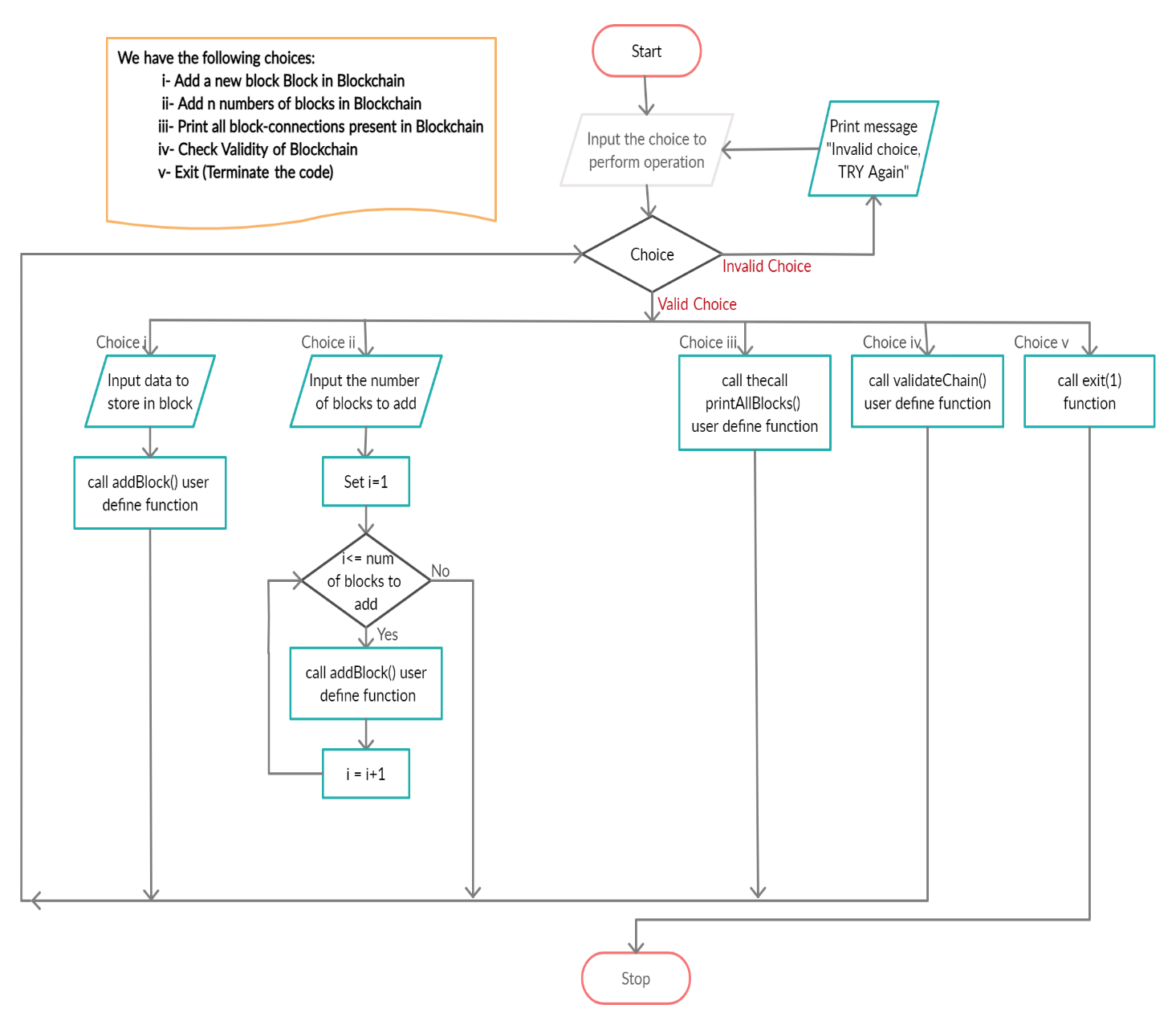
**Pictorial form of system requirements**



**DESIGN**

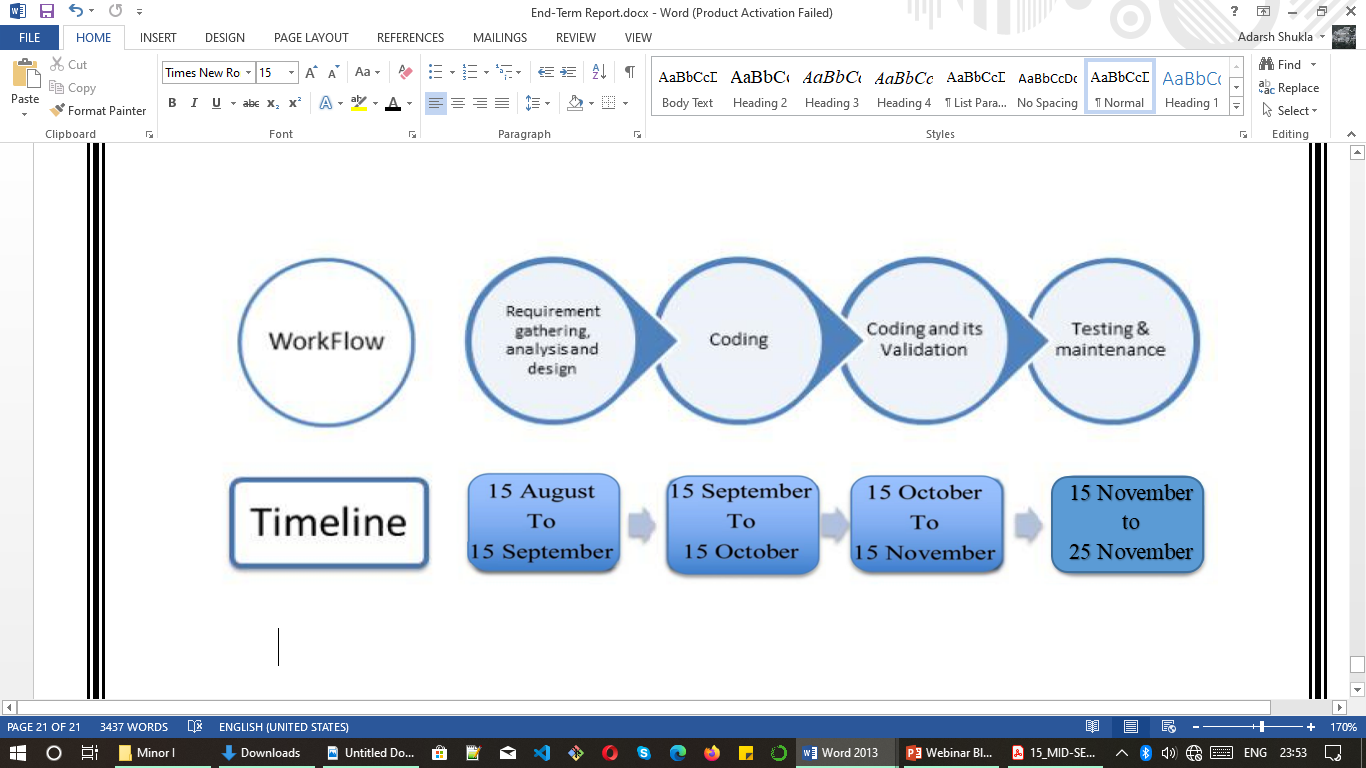
* **Data Flow Diagram (Flow Chart)**

Data flow diagrams like flow chart shows the flow of execution of whole project. I design an interactive flow chart below.



* **Pert Chart**

Pert chart shows the process or can say various module completion of particular project with respect to time. I designed my following pert chart in earlier time before starting the project in month of August.



**ALGORITHMS**

* Main implemented program’s algorithm

**Step 1:** Start

**Step 2 :** Input the choice to perform operation in blockchain

We have following choices:

i- Add a new block Block in Blockchain

ii- Add n numbers of blocks in Blockchain

iii- Print all blocks-connection in Blockchain

iv- Check Velidity of Blockchain

v- Exit (Terminate the code)

**Step 3:** If Choose (i) choice than

a- Input data to store in block

b- call addBlock() user define function than goto *(Step 2)*

**Step 4:** If Choose (ii) choice than

a- Input the number of blocks that you want to add in Blockchain

b- Iterate a for loop (number of blocks you want to add) times

c- In every iteration call addBlock() user define function

d- After for loop iterations goto (Step 2)

**Step 5:** If Choose (iii) choice than

1. call printAllBlocks() user define function than goto *(Step 2)*

**Step 6:** If Choose (iv) choice than

1. call verifyChain() user define function than goto *(Step 2)*

**Step 7:** If Choose (v) choice than

1. exit() function call and terminate the program

**Step 8:** Stop

* **Algorithms of indivisual modules / functions of project**
* **Algo of addBlock() function with block data**

**Step 1:** Start

**Step 2:** Recive block data by user

**Step 3:** Check,

a- if Blockchain is empty that create first block

i.e. Genesus Block

else Add a new block in existing Blockchain

b- Calculate the hash value for this perticular block

c- Store the block data in perticular block

d- Manage addresses of blocks according to concept of linked list

**Step 4:** Stop

* **Algo for printAllBlocks()**

**Step 1:** Start

**Step 2:** Iterate every blocks in Blockchain (nodes in linked list)

Print all the blocks parameters (Block Hash Value, Previous block hash value, Block data, address of next block)

**Step 3:** Stop

* **Algo for validateChain()**

**Step 1:** Start

**Step 2:** Check , Blockchain is empty or not

**Step 3:** if blockchain is empty than

print the message "Hey, Blockchain is empty! Please try

after adding some block!"

else

print message of velidity "Adarsh's Blockchain is valid !!"

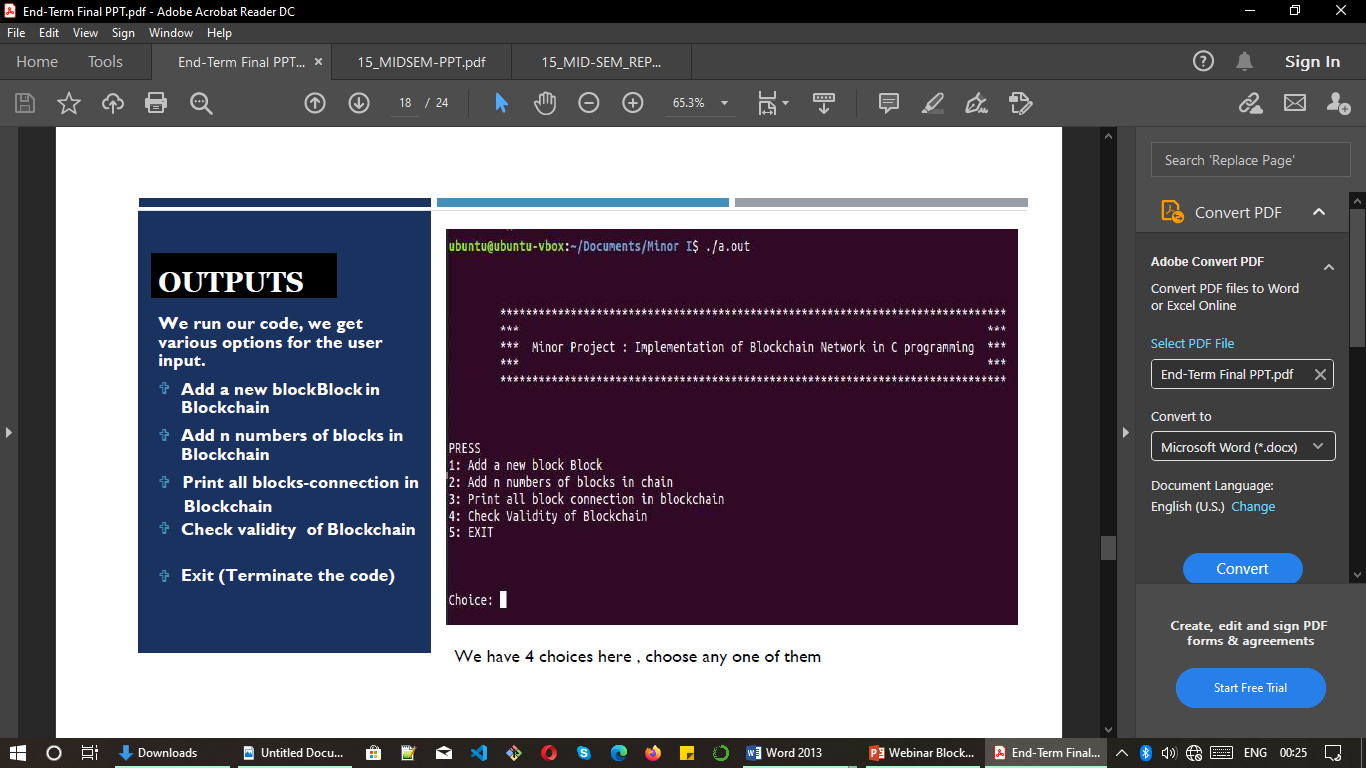
**Step 4:** Stop

**Key Points with respect to ‘Algorithm’**

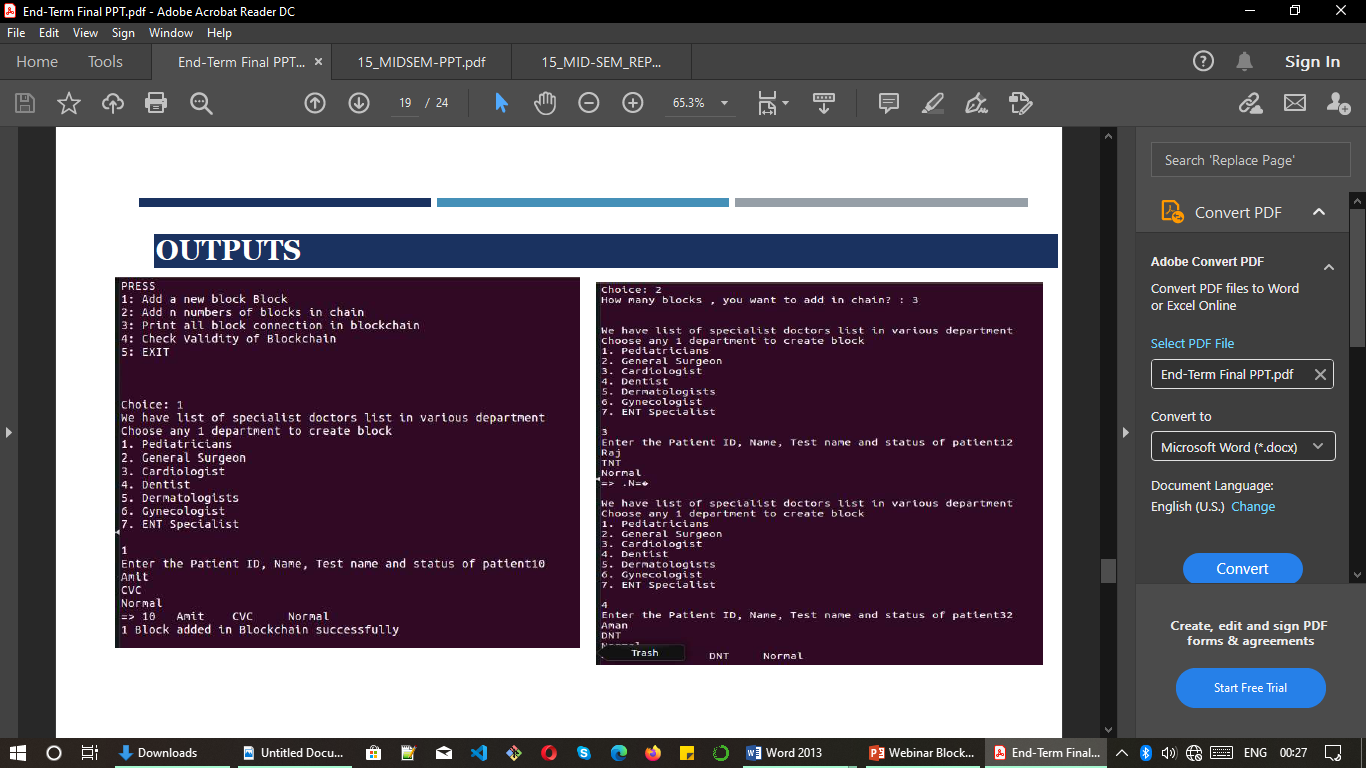
* Algorithms shows the logic of code step by step.
* Algorithms contains set of statements/instructions.
* Algorithm can’t explain the time and apace complexity of program.
* In my project, there are many modules or sub-routines: main function and other user define methods. I write algorithms of all the modules individually above.

**OUTPUT SCREENSHOTS**

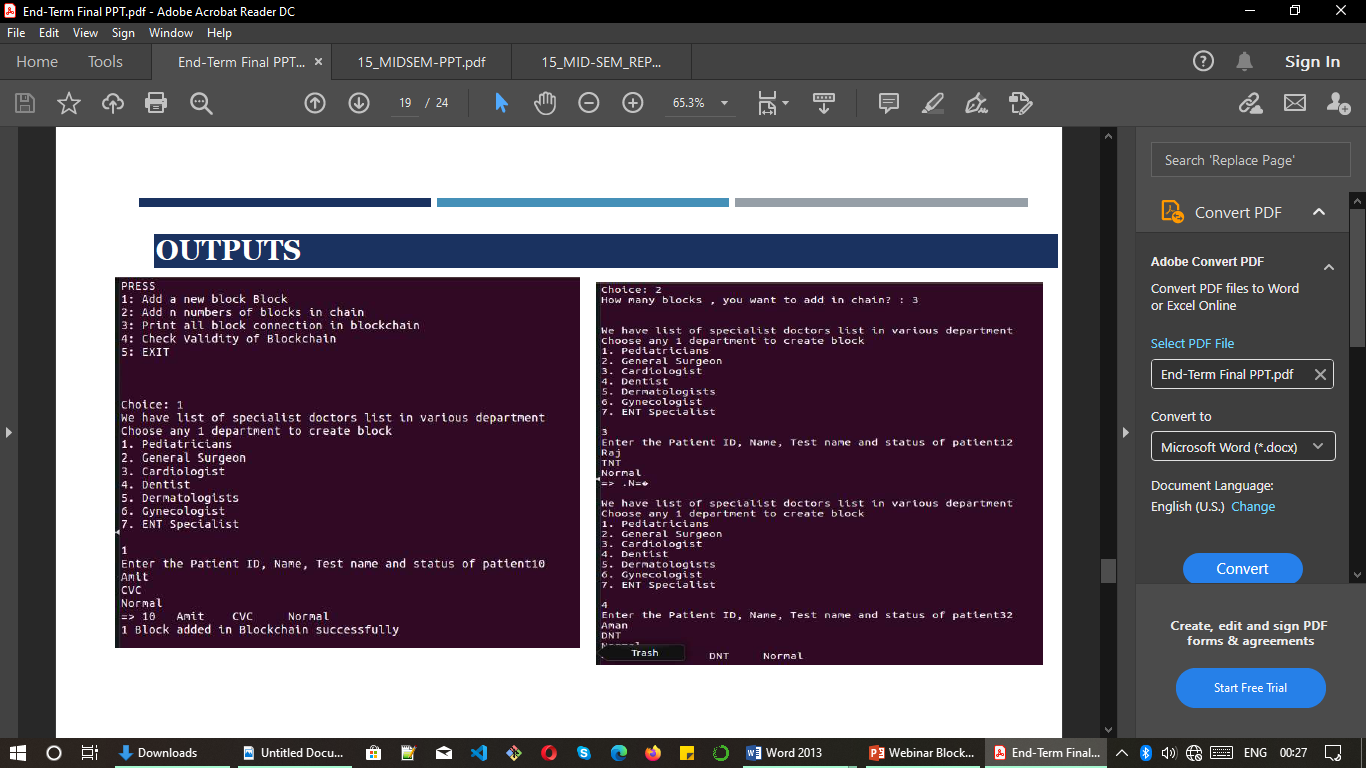
**(1)**



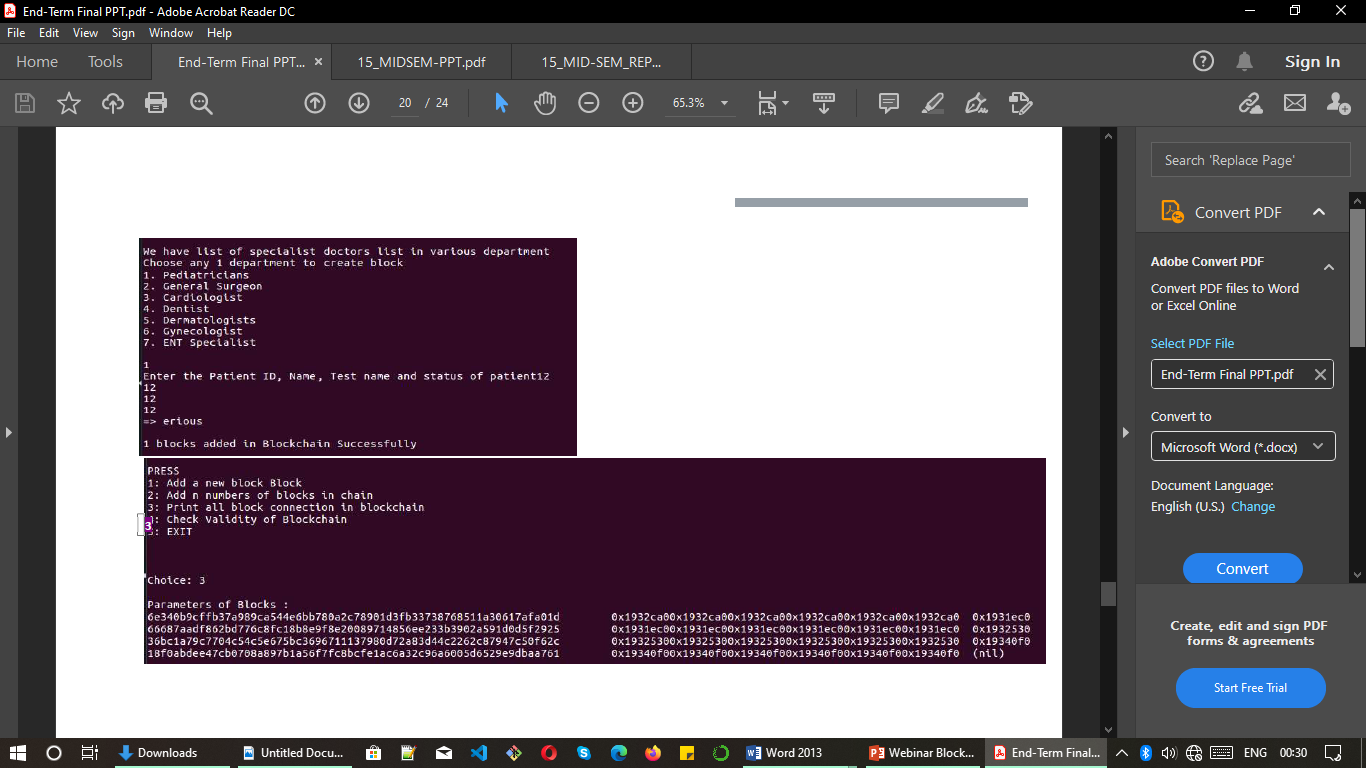
**(2)**



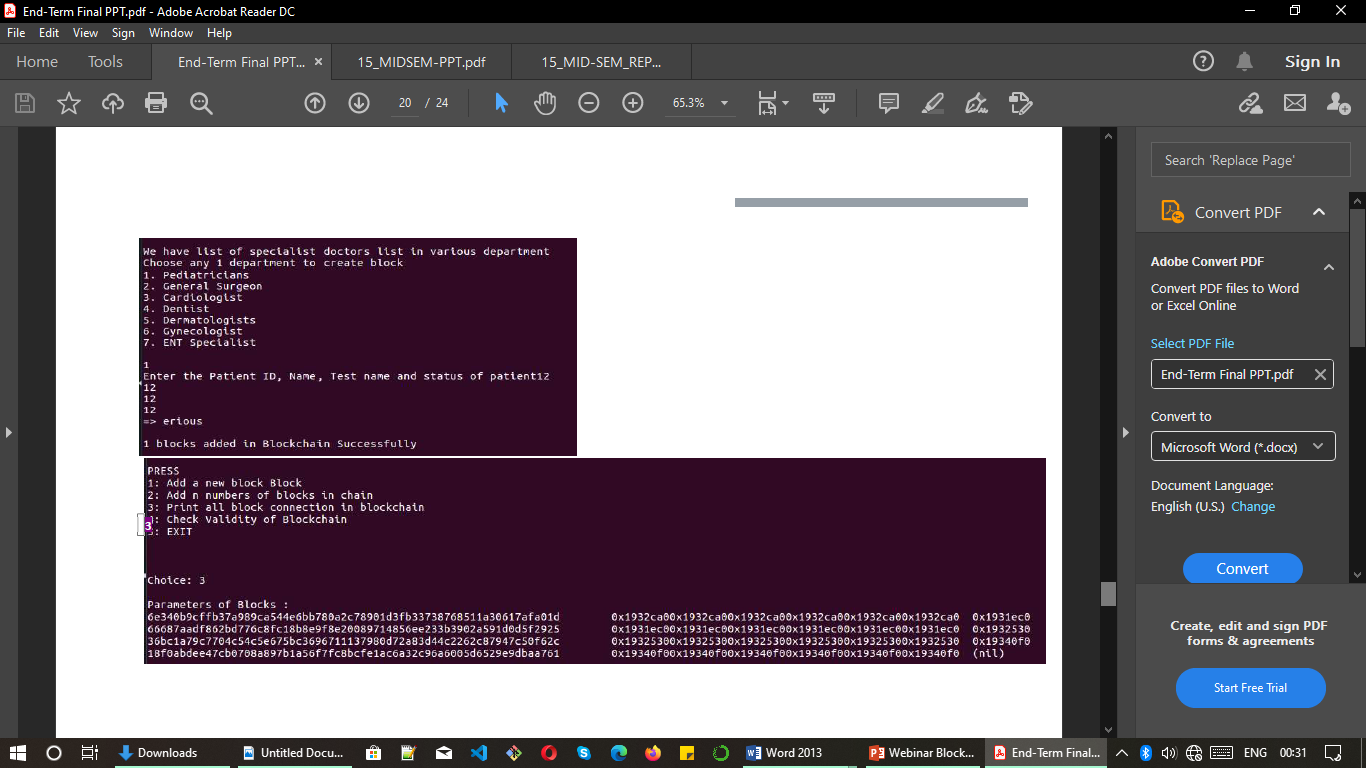
**(3)**



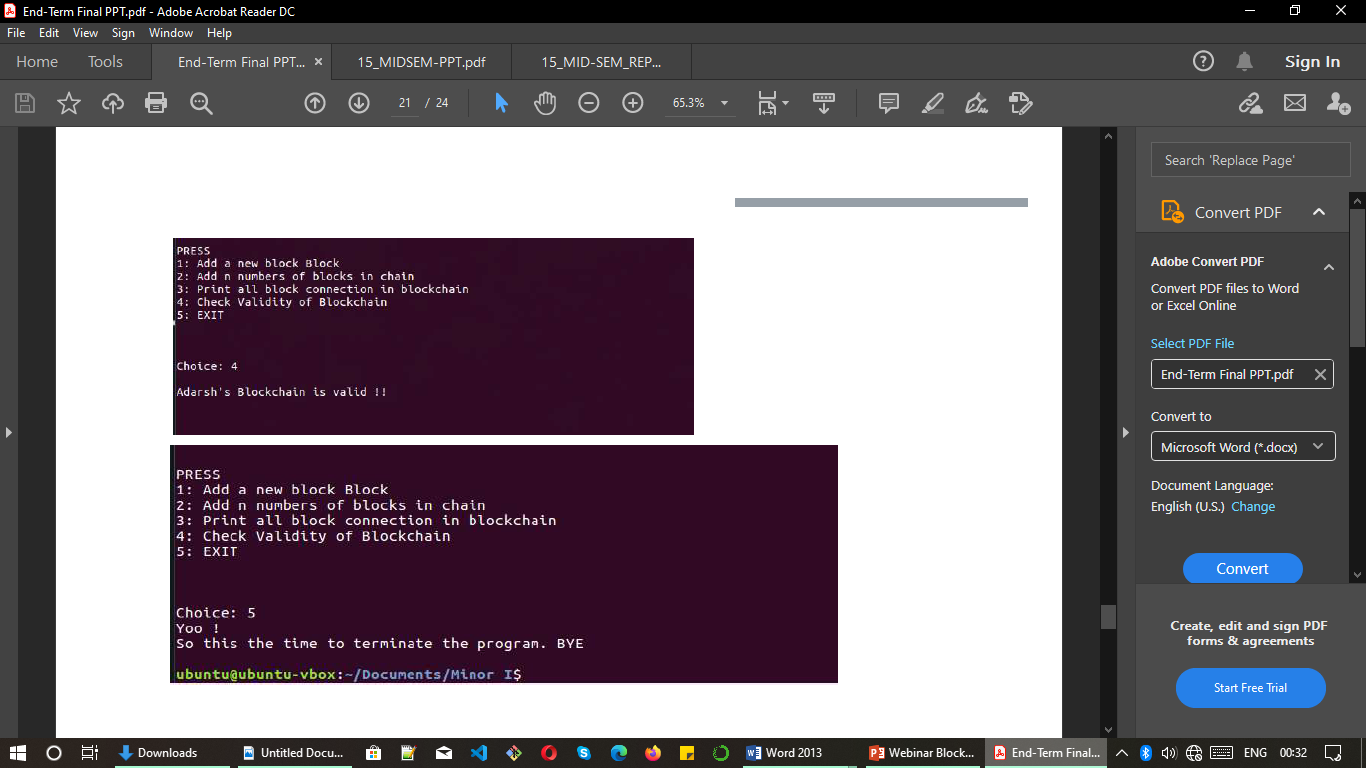
**(4)**



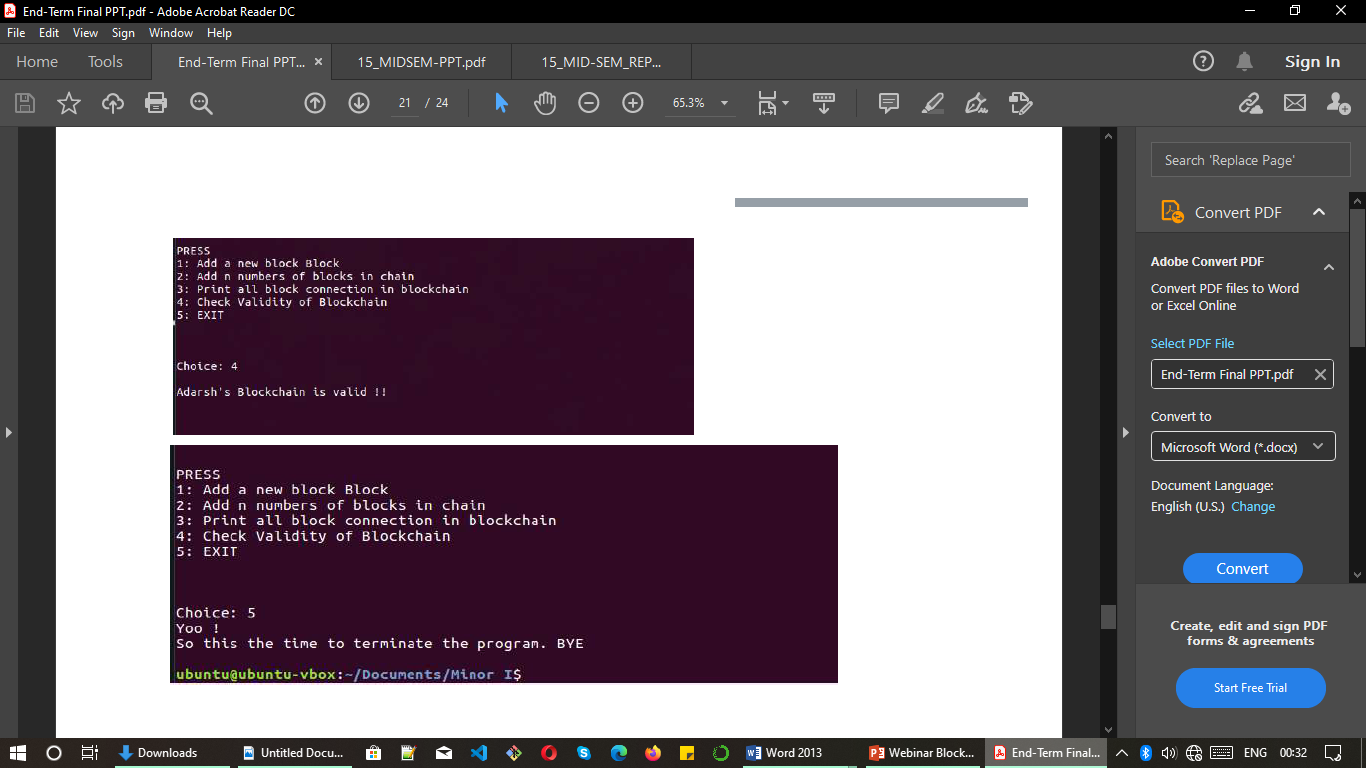
**(5)**



**(6)**

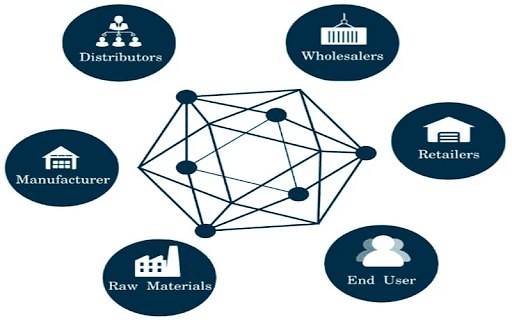


**(7)**

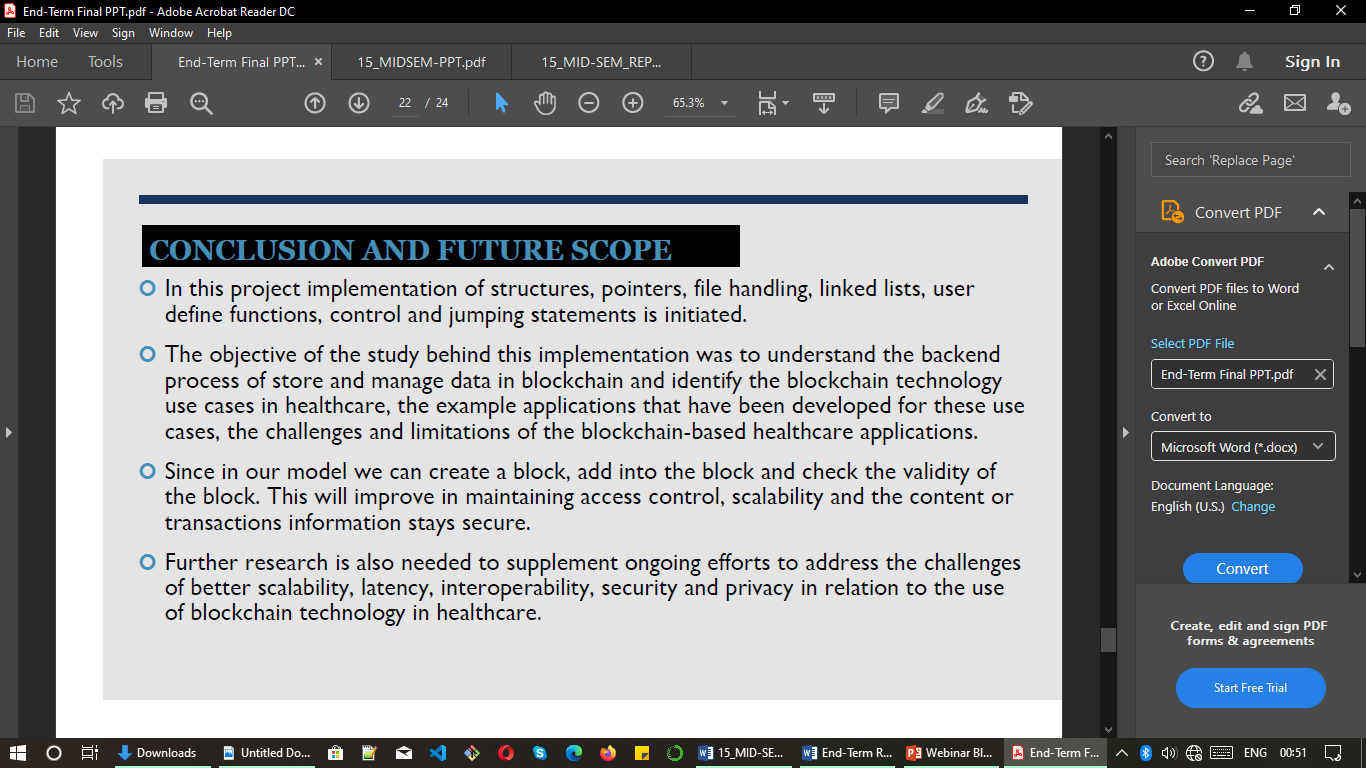


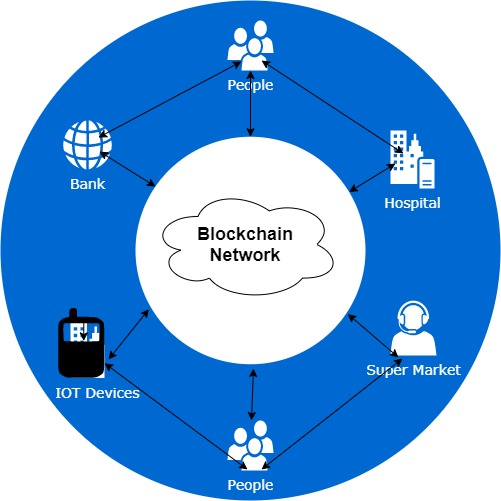
**CONCLUSSION AND FUTURE SCOPE**

Our blockchain model implementation in C language is successful. In this project implementation of structures, pointers, linked lists, user define functions, control and jumping statements was initiated. File handling concept is also applied. We know that since the starting of the era when blockchain technology was launched via bitcoin, it was evolving into a general purpose technology with use cases in many industries including healthcare. The objective of the study behind this model was to identify the blockchain technology use cases in healthcare, the example applications that have been developed for these use cases, the challenges and limitations of the blockchain based healthcare applications, the current approaches employed in developing these applications and areas for future research. Since in our model we can create a block, add into the block and check the validity of the block. This will improve in maintaining access control, scalability and the content or transactions information stays secure. Further research is also needed to supplement ongoing efforts to address the challenges of better scalability, latency, interoperability, security and privacy in relation to the use of blockchain technology in healthcare.



**Key point summery of project**



****

**REFERENCES**

[1] McAndrews, J. (1997). Network issues and payment systems. Federal Reserve Bank of Philadelphia Business Review, December.

[2] Agarwal, S., Khapra, M., Menezes, B., & Uchat, N. (2007, December). Secu- rity issues in mobile payment systems. In Proceedings of ICEG 2007: The 5th In- ternational Conference on E-Governance (pp. 142-152).

[3] Lorenz, G. W. (1996). Electronic stored value payment systems, market posi- tion, and regulatory issues. Am. UL Rev., 46, 11

[4] Crosby, M., Pattanayak, P., Verma, S., & Kalyanaraman, V. (2016). Blockchain technology: Beyond bitcoin. Applied Innovation, 2(6-10), 71.

[5] Gupta, S. S. (2017). Blockchain. John Wiley & Sons, Inc.

[6] Risius, M., & Spohrer, K. (2017). A blockchain research framework. Business & Information Systems Engineering, 59(6), 385-4093

[7] Zhang, Y., Deng, R. H., Liu, X., & Zheng, D. (2018). Blockchain-based efficient and robust fair payment for outsourcing services in cloud computing. Information Sciences, 462, 262-277.

[8] Khalil, R., & Gervais, A. (2017, October). Revive: Rebalancing off-blockchain payment networks. In Proceedings of the 2017 ACM SIGSAC Conference on Com- puter and Communications Security (pp. 439-453).

[9] Ølnes, S., Ubacht, J., & Janssen, M. (2017). Blockchain in Government: Bene- fits and implications of distributed ledger technology for information sharing.

[10] Chan, P. M. W., Lee, J. J. S., & Haldenby, P. A. J. (2019). U.S. Patent No. 10,282,711. Washington, DC: U.S. Patent and Trademark Office.

[11] Nguyen, G. T., & Kim, K. (2018). A Survey about Consensus Algorithms Used in Blockchain. Journal of Information processing systems, 14(1).

[12] Chaudhry, N., & Yousaf, M. M. (2018, December). Consensus algorithms in blockchain: comparative analysis, challenges, and opportunities. In 2018 12th In- ternational Conference on Open Source Systems and Technologies (ICOSST) (pp. 54-63). IEEE.

[13] Zoican, S., Vochin, M., Zoican, R., & Galatchi, D. (2018, November). Block- chain and consensus algorithms in the Internet of Things. In 2018 International Symposium on Electronics and Telecommunications (ISETC) (pp. 1-4). IEEE.

[14] Gramoli, V. (2020). From blockchain consensus back to byzantine consen- sus. Future Generation Computer Systems, 107, 760-769.

[15] Kirda, E., Kruegel, C., Vigna, G., & Jovanovic, N. (2006, April). Noxes: a client-side solution for mitigating cross-site scripting attacks. In Proceedings of the 2006 ACM symposium on Applied computing (pp. 330-337).

[16] Bisht, P., & Venkatakrishnan, V. N. (2008, July). XSS-GUARD: precise dy- namic prevention of cross-site scripting attacks. In International Conference on De- tection of Intrusions and Malware, and Vulnerability Assessment (pp. 23- 43). Springer, Berlin, Heidelberg.

[17] Jain, S., Tomar, D. S., & Sahu, D. R. (2012). Detection of Javascript Vulnera- bility At Client Agen. International Journal of Scientific & Technology Re- search, 1(7), 36-41.

[18] Po-Wei Chen , Bo-Sian Jiang , Chia-Hui Wang . Blockchain-based payment collection supervision system using pervasive Bitcoin digital wallet. 2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob).

[19] Christophe van Cauwenberghe.Blockchain and payments:Lessons learned and future prospects. Societe Generale.

[20] Celine Chen and Shawn Hayashikawa. Easier and Faster Payments with Blockchain. Societe Generale.

[21] Chowdhury, M. J. M., Ferdous, M. S., Biswas, K., Chowdhury, N., & Muthuk- kumarasamy, V. (2020). A survey on blockchain-based platforms for IoT use- cases. Knowledge Eng. Review, 35, e19.\_15

[22] Kiayias, A., Russell, A., David, B., & Oliynykov, R. (2017, August). Ourobo- ros: A provably secure proof-of-stake blockchain protocol. In Annual International Cryptology Conference (pp. 357-388). Springer, Cham.\_19

[23] Duong, T., Fan, L., & Zhou, H. S. (2016). 2-hop blockchain: Combining proof- of-work and proof-of-stake securely. Cryptology ePrint Archive, Report 2016/716.\_20

[24] Kumar, A., & Jain, S. (2019, July). Proof of Game (PoG): A Game Theory Based Consensus Model. In International Conference on Sustainable Communica- tion Networks and Application (pp. 755-764). Springer, Cham.\_21

[25] Erin English. Can Blockchain Help Reduce the Financial Industry's Cyber Risk?.MMC Publication 2018