SPL-1 Project Report, 2022

BlockChainEd

SE: 305

Submitted by

Nusrat Jahan Lia

BSSE Roll No. : 1306

BSSE Session:2020-2021

Supervised by

Dr. Ahmedul Kabir

Designation: Associate Professor

Institute of Information Technology

Institute of Information Technology

University of Dhaka

21-05-2023

Table of Contents

Text: Times New Roman 12, Heading: Bold, Heading type from 1 to 3. Each heading will start with a number indicating its level. Page Type: A4 Left and Right Margin: 1 Inch Top and Bottom margin: 0.5 Inch

1. Introduction

BlockChainEd is an educational project that empowers users to create and explore their own blockchain networks. With an intuitive app, users can easily generate wallets, conduct transactions, assign miners, and examine decentralized information. The application validates transactions, executes proof of work, and seamlessly integrates verified blocks into the blockchain. Additionally, BlockChainEd offers a unique feature that allows users to simulate attacks on their blockchain, enabling them to modify information within a valid block. However, the app promptly safeguards against such attacks by recalculating hashes and breaking the chain from the corrupted block. BlockChainEd serves as an interactive platform that not only educates users about blockchain technology but also allows them to experience its practical implementation.

1.1. Background Study

To implement the "BlockChainEd" project, I had to gather the following study and background knowledge:

1. Blockchain Technology: Understanding the fundamental concepts and principles of blockchain technology, including decentralized consensus mechanisms, distributed ledger systems, blocks, and transactions.
2. Cryptography: Knowledge of cryptographic techniques used in blockchain systems, such as public key cryptography, elliptic curve cryptography (ECC), and digital signatures. Understanding the concepts of private keys, public keys, and how they are used to secure transactions and wallets.
3. Wallet Creation: Learning about the process of generating public-private key pairs for wallet addresses, creating wallets, and managing key pairs securely. Understanding how wallet addresses are used to send and receive transactions.
4. Transactions and Mining: Understanding how transactions are created, signed, and propagated in a blockchain network. Studying the process of appointing miners to validate transactions, perform proof-of-work (PoW) calculations, and add verified blocks to the blockchain.

5. Consensus Mechanisms: Researching different consensus mechanisms used in blockchain networks, such as PoW or proof-of-stake (PoS). Understanding how consensus algorithms ensure the agreement and security of the blockchain.
6. Blockchain Data Structure: Studying the data structure of a blockchain, including the linkage between blocks using cryptographic hashes, maintaining the integrity of the chain, and ensuring immutability.
7. Information Inspection: Exploring methods to inspect and retrieve information stored in a decentralized environment. Understanding how to access and verify transaction details, block contents, and other relevant data.
8. Blockchain Attacks: Researching various types of attacks that can occur in a blockchain network, such as double-spending attacks. Understanding the vulnerabilities and countermeasures to prevent such attacks.
9. SHA-256 Hashing Algorithm: Learning about the SHA-256 algorithm used for hashing in blockchain systems. Understanding how it ensures the integrity and security of transactions and blocks.
10. Software Development: Acquiring programming skills and knowledge of Object Oriented programming in C++ to implement the functionalities of the "BlockChainEd" application, including wallet creation, transaction handling, mining, block verification, and attack prevention.

By gathering the above study and background knowledge, one would have the necessary foundation to implement the "BlockChainEd" project successfully.

1.2. Challenges

Implementing the "BlockChainEd" project comes with several challenges:

1. Complexity: Blockchain technology involves intricate concepts like consensus algorithms, cryptographic techniques, and decentralized data structures. Understanding and implementing these concepts correctly can be challenging, especially for developers who are new to blockchain.
2. Security: Security is a crucial aspect of blockchain systems. Ensuring the secure generation and storage of private keys, implementing robust authentication and encryption mechanisms, and protecting against potential attacks require careful attention and expertise.
3. Performance and Scalability: As the blockchain grows in size and the number of transactions increases, maintaining optimal performance and scalability becomes challenging. Efficient transaction processing, block validation, and network scalability need to be considered during the development process.
4. Consensus Mechanisms: Implementing a consensus mechanism, such as proof-of-work (PoW) or proof-of-stake (PoS), involves addressing challenges related to block validation,

miner selection, and network coordination. Each consensus algorithm has its own complexities that need to be understood and implemented correctly.

5. Blockchain Attacks: Anticipating and preventing various attacks, such as 51% attacks, double-spending attacks, or Sybil attacks, requires a deep understanding of the vulnerabilities and countermeasures specific to blockchain systems. Implementing robust security measures to protect the integrity of the blockchain is crucial.
6. User Experience: Designing a user-friendly interface that allows users to create wallets, make transactions, and inspect blockchain information intuitively can be a challenge. Balancing simplicity with the complex underlying technology is essential to ensure a positive user experience.
7. Testing and Debugging: Blockchain applications require thorough testing to identify and fix potential bugs or vulnerabilities. Ensuring the reliability and stability of the application, particularly in a decentralized environment, can be challenging and time-consuming.
8. Integration and Interoperability: Integrating the blockchain application with other systems or platforms, such as cryptocurrency wallets or third-party services, may present challenges. Ensuring compatibility and seamless interaction between different components can be complex.

Overcoming these challenges requires a combination of deep technical knowledge, careful planning, and continuous testing and improvement. It is important to stay updated with the latest advancements and best practices in blockchain development to address these challenges effectively.

2. Project Overview

BlockChainEd is an innovative project that I have been working on, aimed at empowering users to create their own blockchain networks. With BlockChainEd, users have the ability to create wallets, make transactions, and even appoint miners within their personalized blockchain environment. The application incorporates various essential features such as transaction verification, proof of work, and the seamless addition of verified blocks to the blockchain.

One of the key functionalities of BlockChainEd is the ability for users to inspect and explore the information within their decentralized environment. This provides users with a comprehensive understanding of the blockchain's inner workings and enhances their overall control over the system.

To ensure the utmost security and privacy, BlockChainEd leverages public-key cryptography, with a specific focus on Elliptic Curve Cryptography (ECC). This cryptographic technique enables secure key generation, encryption, and digital signature operations, ensuring the integrity and confidentiality of the user's transactions.

Furthermore, BlockChainEd implements the widely adopted SHA256 hashing algorithm. This algorithm plays a vital role in verifying the integrity of the data stored within the blockchain by

generating a unique hash for each block. This ensures that any modifications to the information contained within a block will be immediately detected, safeguarding the integrity of the blockchain.

An intriguing feature of BlockChainEd is the ability for users to test the resilience of their own blockchain by simulating an attack scenario. Users can attempt to change the information within a valid block, but BlockChainEd actively defends against such attacks. The application achieves this by recalculating the hash and subsequently breaking the chain from the corrupted block, ensuring the security and immutability of the blockchain.

In summary, BlockChainEd is an ambitious project that allows users to create, manage, and explore their own blockchain networks. By incorporating public-key cryptography using ECC and implementing the SHA256 hashing algorithm, BlockChainEd offers a secure and robust environment for users to engage in transactions while maintaining the integrity and privacy of their data.

3. User Manual

This user manual provides step-by-step instructions on how to obtain the BlockChainEd application from GitHub and run the RunBlockChainEd.cpp file. Please follow the guidelines below:

Prerequisites:

1. C++ Compiler: Ensure that you have a C++ compiler installed on your system, such as GCC or Clang.
2. Code Editor/IDE: Have a code editor or integrated development environment (IDE) installed to open and edit the RunBlockChainEd.cpp file.

Instructions:

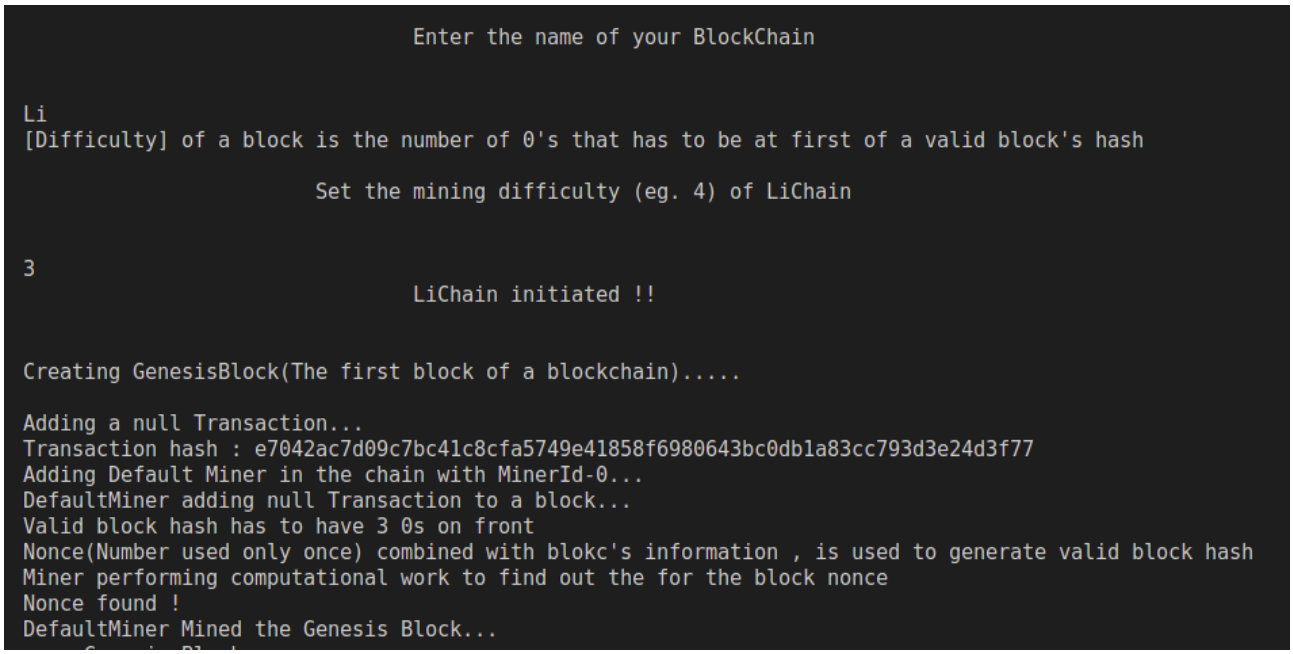
1. Visit the BlockChainEd repository on GitHub at https://github.com/NusRAT-LiA/Blockchain_in_Cpp-SPL-1 .
2. On the GitHub repository page, click on the "Code" button and select the option to clone the repository. Copy the provided repository URL.
3. Open the Terminal/Command Prompt: Launch your system's terminal or command prompt to execute commands.
4. In the terminal or command prompt, navigate to the desired directory where you want to clone the BlockChainEd repository. Use the following command to clone the repository:

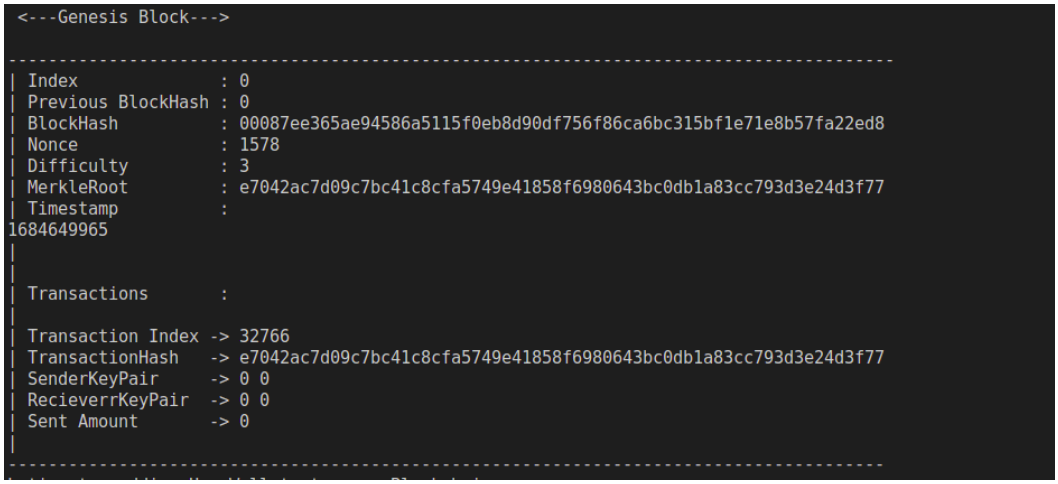
   git clone https://github.com/NusRAT-LiA/Blockchain_in_Cpp-SPL-1
5. Compile the RunBlockChainEd.cpp file using the C++ compiler. Open your terminal or command prompt and navigate to the UserInterface directory where the RunBlockChainEd.cpp file is located. Then, run the following command:

   g++ RunBlockChainEd.cpp -o BlockChainEd

6. Choose from options give :

7. Enter your desired name for your blockchain and select difficulty level . A genesis block will be initiated .





8. Create desired amount of Wallets and Transactions using wallet's information .

```
Let's star adding UserWallets to your Blockchain ......

How many wallets do you want do add ?

2
Enter UserName   : lia
Allocate Balance :30
Rremember lia's  Private Key to verify transactions requested from this  wallet !!  : 28034
Account address :  1841fdc0ada09be81ec2c3d5a7054e7bec4becdaf42cfc1ab4a79a6213e1a650 created!

Public Key is   : 464603 -301596
Enter UserName   : tump
Allocate Balance :20
Rremember tump's  Private Key to verify transactions requested from this  wallet !!  : 107667
Account address :  c5a1b5a0ff2d3da3f5d2c041491b7712b765bd8a4552dc763bdc5d84fa5faa8b created!

Public Key is   : 518577 -215243
Let's make Transactions !!...

How many Transactions do you want to make ?

1
Sender Public Key                       : 464603 -301596

Reciever Public Key                     : 518577 -215243

Amount to be sent                       : 18

Digital Signature(Sender's PrivateKey)  : 28034
Offered Transaction fee                 : 3
Transaction hash : 85651d9fbb7bf1ef233fcc1e9f93858de2c2895cbf697fb58b6e62b4388b6a8f
You already have a default miner with MinerID 0

Do you want to add more miners ?(YES or NO)

no
Let's start mining Blocks !
```

9. Appoint miner to mine the block

```
. . .

Enter Miner ID to appoint miner for mining block   :

0
Miner ID0 Collecting Transactions from the network with higher fees
Block being created with ..
Tx hash            : 85651d9fbb7bf1ef233fcc1e9f93858de2c2895cbf697fb58b6e62b4388b6a8f
Network difficulty : 3
Previous blockhash : 00087ee365ae94586a5115f0eb8d90df756f86ca6bc315bf1e71e8b57fa22ed8
Miner ID0  started verifying block's transactions...
Transaction: 85651d9fbb7bf1ef233fcc1e9f93858de2c2895cbf697fb58b6e62b4388b6a8f verified!
Miner ID0 started mining block
Valid block hash has to have 3 0s on front
Nonce(Number used only once) combined with blokc's information , is used to generate valid block hash
Miner performing computational work to find out the for the block nonce
Nonce found !
Block mined with nonce 138608736
Transaction 85651d9fbb7bf1ef233fcc1e9f93858de2c2895cbf697fb58b6e62b4388b6a8fsuccessful !
New balance of sender wallet1841fdc0ada09be81ec2c3d5a7054e7bec4becdaf42cfc1ab4a79a6213e1a650 : 9
New balance of reciever walletc5a1b5a0ff2d3da3f5d2c041491b7712b765bd8a4552dc763bdc5d84fa5faa8b : 38
Transaction fee 3 added to Miner ID 0's balance

Block added to the chain successfully !
```

10. Verified block will be added to your chain

```
Here's your chain !-------------------------------------------------------------------------------------------
| Index              : 0
| Previous BlockHash : 0
| BlockHash          : 00087ee365ae94586a5115f0eb8d90df756f86ca6bc315bf1e71e8b57fa22ed8
| Nonce              : 1578
| Difficulty         : 3
| MerkleRoot         : e7042ac7d09c7bc41c8cfa5749e41858f6980643bc0db1a83cc793d3e24d3f77
| Timestamp          :
1684649965
|
|
| Transactions       :
|
| Transaction Index -> 32766
| TransactionHash   -> e7042ac7d09c7bc41c8cfa5749e41858f6980643bc0db1a83cc793d3e24d3f77
| SenderKeyPair     -> 0 0
| RecieverrKeyPair  -> 0 0
| Sent Amount       -> 0
|
-------------------------------------------------------------------------------------
^
|
|
-------------------------------------------------------------------------------------------
| Index              : 1
| Previous BlockHash : 00087ee365ae94586a5115f0eb8d90df756f86ca6bc315bf1e71e8b57fa22ed8
| BlockHash          :
| Nonce              : 138608736
| Difficulty         : 3
| MerkleRoot         :
| Timestamp          :
94128144310343
|
|
| Transactions       :
|
| Transaction Index -> 1
| TransactionHash   -> 85651d9fbb7bf1ef233fcc1e9f93858de2c2895cbf697fb58b6e62b4388b6a8f
| SenderKeyPair     -> 464603 -301596
| RecieverrKeyPair  -> 518577 -215243
| Sent Amount       -> 18
|
-------------------------------------------------------------------------------------
```

11 . Chose to attack

```
Choose from options :

1 . Create  BlockChain

2 . Inspect BlockChain

3 . Attack BlockChain

5 . Exit

2
```

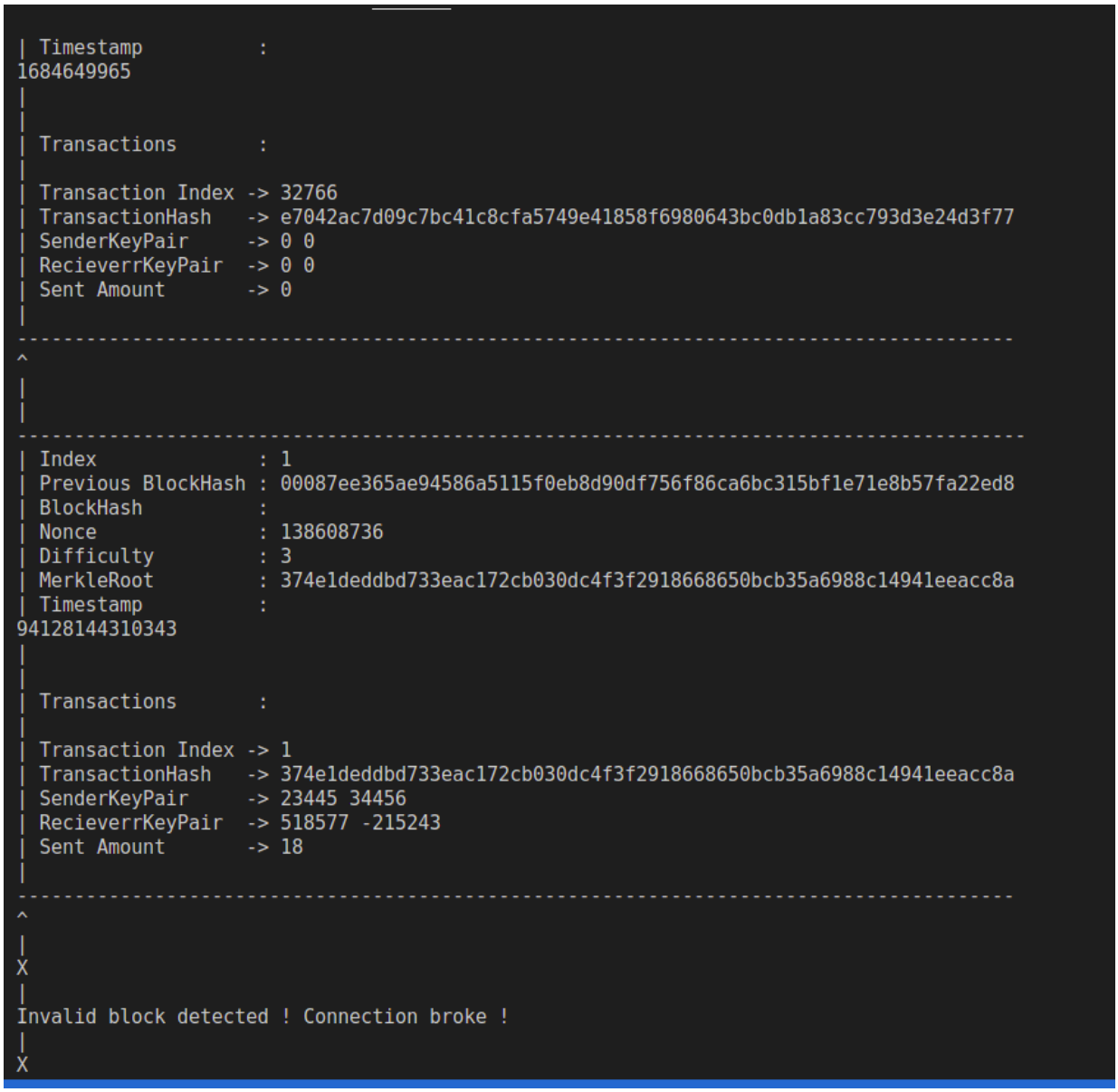
12. Enter the block's index and transaction index you want access to attack

```
Enter the index of valid Block you want access to :

1
The block :


-------------------------------------------------------------------------------------
| Index              : 1
| Previous BlockHash : 00087ee365ae94586a5115f0eb8d90df756f86ca6bc315bf1e71e8b57fa22ed8
| BlockHash          :
| Nonce              : 138608736
| Difficulty         : 3
| MerkleRoot         :
| Timestamp          :
94128144310343
|
|
| Transactions       :
|
| Transaction Index -> 1
| TransactionHash   -> 85651d9fbb7bf1ef233fcc1e9f93858de2c2895cbf697fb58b6e62b4388b6a8f
| SenderKeyPair     -> 464603 -301596
| RecieverrKeyPair  -> 518577 -215243
| Sent Amount       -> 18
|
-------------------------------------------------------------------------------------
Enter the index(starting from 1) of valid Transaction in the block you want access to :

1
Choose from options :

1 . Change Sender's   Public Key

2 . Change Reciever's Public Key

3 . Change Sender's   Private Key

4 . Change sentAmout

5 . Exit

1
```

13. Change desired value

```
X :

23445
Y :

34456
Choose from options :

1 . Change Sender's   Public Key

2 . Change Reciever's Public Key

3 . Change Sender's   Private Key

4 . Change sentAmout

5 . Exit

5
```

14. Invalid Block Detected !



15. Continue to create and explore !

4. Conclusion

In conclusion, the "BlockChainEd" project is a complex and challenging endeavor that requires a comprehensive understanding of blockchain technology, cryptography, consensus mechanisms, and security measures. The implementation of functionalities such as wallet creation, transaction handling, mining, block verification, and information inspection demands expertise in various programming languages and software development skills.

Throughout the project, several hurdles need to be overcome. These challenges include grappling with the intricacies of blockchain technology, ensuring robust security measures to protect against

attacks, addressing performance and scalability issues, designing a user-friendly interface, and conducting rigorous testing and debugging.

Despite these challenges, successfully implementing the "BlockChainEd" project can yield significant benefits. Users will be empowered to create their own blockchain, perform transactions securely through public key cryptography and ECC, and gain insights into decentralized environments. By allowing users to simulate and prevent attacks on their blockchain, the project facilitates a deeper understanding of blockchain vulnerabilities and countermeasures.

Overall, the "BlockChainEd" project serves as an educational tool, providing individuals with a hands-on experience in blockchain development. By gathering the necessary study and background knowledge and overcoming the associated challenges, developers can contribute to the advancement and adoption of blockchain technology, a transformative force with the potential to revolutionize various industries and foster a more decentralized and secure digital landscape.

5. Appendix

Here are some additional resources that can provide further information and guidance on various aspects related to the "BlockChainEd" project:

1. **Bitcoin Whitepaper**: The original whitepaper titled "Bitcoin: A Peer-to-Peer Electronic Cash System" by Satoshi Nakamoto, which introduced the concept of blockchain and the underlying technology behind it. Available at: https://bitcoin.org/bitcoin.pdf
2. **Ethereum Documentation**: The official documentation for the Ethereum blockchain platform, providing detailed information on smart contracts, decentralized applications (dApps), and the Ethereum Virtual Machine (EVM). Available at: https://ethereum.org/developers/
3. **Blockchain Basics**: A comprehensive guide by ConsenSys that covers the foundational concepts of blockchain technology, including consensus mechanisms, cryptographic techniques, and decentralized networks. Available at: https://consensys.net/blockchain-basics/
4. **Mastering Blockchain**: A book by Imran Bashir that explores advanced topics in blockchain technology, including consensus algorithms, privacy, and interoperability. It provides practical examples and code samples for implementing blockchain solutions. Available on various online platforms.
5. **Cryptozombies**: An interactive tutorial that teaches Solidity programming for Ethereum smart contracts through a fun, gamified approach. It covers key concepts, such as deploying contracts, creating tokens, and implementing game logic. Available at: https://cryptozombies.io/
6. **Blockchain Security**: A book by Ghassan Karame, Elli Androulaki, and Srdjan Capkun that delves into the security aspects of blockchain systems, including cryptographic primitives, secure key management, and vulnerability analysis. Available on various online platforms.

7. **Bitcoin and Cryptocurrency Technologies**: A book by Arvind Narayanan, Joseph Bonneau, Edward Felten, Andrew Miller, and Steven Goldfeder that provides a comprehensive introduction to cryptocurrencies and blockchain technology. It covers various technical and non-technical aspects of the field. Available on various online platforms.
8. **StackExchange Cryptography**: An online community where experts and enthusiasts discuss cryptographic concepts, algorithms, and implementations. It can be a valuable resource for addressing specific questions or concerns related to public key cryptography, ECC, or hashing algorithms. Available at: https://crypto.stackexchange.com/

These resources, combined with continuous learning and exploration, will further enhance your understanding of blockchain technology and assist you in successfully implementing the "BlockChainEd" project.

References

[1] W. Dai, "b-money," http://www.weidai.com/bmoney.txt, 1998.

[2] Software Implementations and Applications of Elliptic Curve Cryptography, Kirill Kultinov/ Wright State University, 2019, 89

[3] Bitcoin: A Peer-to-Peer Electronic Cash System ,Satoshi Nakamoto/2008

[4] H. Massias, X.S. Avila, and J.-J. Quisquater, "Design of a secure timestamping service with minimal trust requirements," In 20th Symposium on Information Theory in the Benelux, May 1999.

[5] S. Haber, W.S. Stornetta, "How to time-stamp a digital document," In Journal of Cryptology, vol 3, no 2, pages 99-111, 1991.

[6] A. Back, "Hashcash - a denial of service counter-measure,"http://www.hashcash.org/papers/hashcash.pdf, 2002.

[7] R.C. Merkle, "Protocols for public key cryptosystems," In Proc. 1980 Symposium on Security and Privacy, IEEE Computer Society, pages 122-133, April 1980