INTERNATIONAL JOURNAL OF NOVEL RESEARCH AND DEVELOPMENT (IJNRD)
© 2026 IJNRD | Volume 11, Issue 5, May 2026 | ISSN: 2456-4184 | IJNRD.ORG

Secure and Scalable Blockchain Data Placement System Using Compression Techniques

PULAPA JYOTHI SAI
Master Of Computer Applications
Ideal College Of Arts & Sciences,
Autonomous, Affiliated To Adikavi Nannaya University - Rajamahendravaram
Kakinada

T. PRIDHVI KRISHNA
Assistant.Prof, Master Of Computer Applications
Ideal College Of Arts & Sciences,
Autonomous, Affiliated To Adikavi Nannaya University - Rajamahendravaram
Kakinada

Dr. V.S.V. DEEPAK
HOD, department Of Computer science
Ideal College Of Arts & Sciences,
Autonomous, Affiliated To Adikavi Nannaya University - Rajamahendravaram
Kakinada

***Abstract*— Blockchain-based edge computing systems are becoming an effective solution for secure and decentralized data trading between producers and consumers. This project develops a secure data trading framework that combines blockchain, relay nodes, and smart contracts to remove dependency on centralized third-party platforms. The proposed extension integrates a compression mechanism with the Rounding-Based Data Placement Algorithm (RDPA) to reduce storage overhead and improve relay node utilization. Data files are encrypted using AES encryption before storage, while blockchain maintains metadata and transaction records for secure verification and transparency. The system also adopts the Proof-of-Data-Trading (PODT) consensus mechanism to achieve stable transactions with lower energy consumption. Experimental analysis shows that the extension model decreases storage cost, improves relay efficiency, and enhances producer revenue when compared with traditional blockchain storage models.**

Keywords*— *Blockchain, Smart Contracts, Data Compression, AES

I. INTRODUCTION

Rapid growth in smart devices, mobile applications, and edge computing technologies has significantly increased the generation and sharing of digital data across networks. Devices such as smartphones, tablets, and IoT sensors continuously produce large volumes of images, videos, documents, and location-based information. Most of this data is stored and managed through centralized cloud platforms, where third-party service providers control storage, transactions, and revenue distribution. Although centralized systems offer convenience and scalability, they also introduce major concerns related to privacy leakage, unauthorized access, data tampering, and unfair profit sharing. Users often lose control over their own data after uploading it to centralized servers.

Blockchain technology has emerged as a reliable solution for secure and decentralized data management because of its transparency, immutability, and distributed architecture. By maintaining transaction records in a tamper-resistant ledger, blockchain enables secure verification and trusted data exchange without depending on centralized authorities. However, traditional blockchain systems face limitations such as high storage overhead, increased transaction cost, and low scalability when handling large multimedia files. Edge computing further improves system performance by processing and storing data closer to users, thereby reducing latency and network congestion. Efficient data placement, secure relay management, and storage optimization have therefore become important research areas in blockchain-based edge environments. These challenges motivate the development of advanced mechanisms that can improve storage efficiency, reduce operational cost, and maintain secure data trading among distributed users.

II. RELATED WORK

Blockchain and edge computing technologies have gained significant attention for enabling secure, decentralized, and efficient data sharing in modern digital environments. Researchers have explored various techniques to improve transaction security, resource allocation, privacy protection, and decentralized communication in distributed systems. Nick Szabo (1997) introduced the concept of smart contracts for automating secure digital agreements without centralized intermediaries, which later became a major foundation for blockchain systems. Castro and Liskov (2002) proposed the Practical Byzantine Fault Tolerance mechanism to ensure reliable communication in distributed networks even in the presence of malicious nodes. Nakamoto (2008) developed Bitcoin using blockchain technology and introduced decentralized ledger management, cryptographic hashing, and consensus-based verification for secure peer-to-peer transactions. Buterin et al. (2014) extended blockchain capabilities through Ethereum by supporting programmable smart contracts and decentralized applications. Lin et al. (2019) proposed a consortium blockchain framework for secure knowledge trading in edge-AI enabled IoT environments, emphasizing decentralized collaboration and incentive management. Zhang et al. (2019) developed a smart contract-based access control mechanism for Internet of Things systems to improve authentication and authorization security. Fang and Lei (2020) analyzed blockchain integration with edge AI computing and discussed challenges related to scalability, privacy, and decentralized learning. Huang et al. (2020) introduced a hybrid Proof-of-Work and Proof-of-Stake incentive mechanism for efficient and low-energy blockchain operations in edge environments. Huang et al. (2023) further proposed a profit-sharing framework for data producers and intermediate parties in blockchain-based edge computing systems to improve relay efficiency and revenue management. Liu et al. (2023) presented a decentralized blockchain framework for secure and authenticated information sharing in zero-trust IoT environments, highlighting the importance of

trusted communication and decentralized security in distributed computing systems.

Table: Summary of Key Literature Contributions and Their Impact on Current Research:

Author	Contribution	Impact on Research
Nick Szabo (1997)	Introduced smart contracts for automatic digital agreements.	Formed the base for secure blockchain transactions and automation.
Castro and Liskov (2002)	Developed PBFT for secure distributed communication.	Improved reliability and fault tolerance in blockchain systems.
Satoshi Nakamoto (2008)	Introduced Bitcoin and decentralized blockchain technology.	Created the foundation for secure and transparent digital transactions.
Vitalik Buterin et al. (2014)	Developed Ethereum with smart contract support.	Enabled decentralized applications and automated blockchain services.
José-Luis de la Rosa et al. (2016)	Applied blockchain for intellectual property protection.	Improved secure ownership verification and digital data sharing.
Xiaodong Lin et al. (2019)	Proposed blockchain-based knowledge trading in IoT systems.	Enhanced secure data sharing in edge computing environments.
Yutaka Zhang et al. (2019)	Developed smart contract-based access control for IoT.	Improved authentication and security in IoT networks.
Jianjun Fang and Kai Lei (2020)	Surveyed blockchain and edge AI computing technologies.	Identified challenges in scalability, privacy, and resource management.
Yong Huang et al. (2020)	Proposed hybrid PoW and PoS incentive mechanism.	Improved energy efficiency and consensus performance.
Yong Huang et al. (2023)	Developed profit-sharing and relay selection framework.	Improved storage efficiency and secure blockchain-based data trading.

III. PROPOSED APPROACH

A secure and decentralized framework is designed for data trading in blockchain-based edge computing environments to eliminate dependency on centralized third-party platforms. The system allows producers to upload and sell digital data items such as images, videos, and documents directly to consumers through blockchain-supported transactions. Initially, both producers and consumers register with the system, and their details are securely maintained using blockchain smart contracts. After authentication, producers can upload data files with related information including description, size, and pricing details.

To ensure confidentiality and data protection, uploaded files are encrypted using the AES encryption algorithm before storage. The SHA-256 hashing algorithm is used to generate digital signatures for maintaining integrity and preventing unauthorized modification of data. Instead of storing complete files in the blockchain, encrypted files are stored in relay nodes that provide high storage capacity and faster processing speed. The blockchain stores only metadata such as file information, digital signatures, upload date, and relay location details, which reduces blockchain storage complexity and improves scalability.

An additional compression mechanism is integrated into the framework to minimize file size before storing data in relay nodes. This process reduces storage cost, improves relay utilization, and prevents relay nodes from becoming full within a short duration. The Rounding-Based Data Placement Algorithm (RDPA) is used to identify suitable relay nodes based on storage availability and bandwidth efficiency, thereby reducing unnecessary relay expenses and improving data placement performance.

For transaction verification, the framework employs the Proof-of-Data-Trading (PODT) consensus mechanism that combines features of Proof-of-Work and Proof-of-Stake to achieve secure, energy-efficient, and stable blockchain operations. Consumers can browse, purchase, and securely download encrypted files, which are decrypted only after successful authorization and transaction validation.

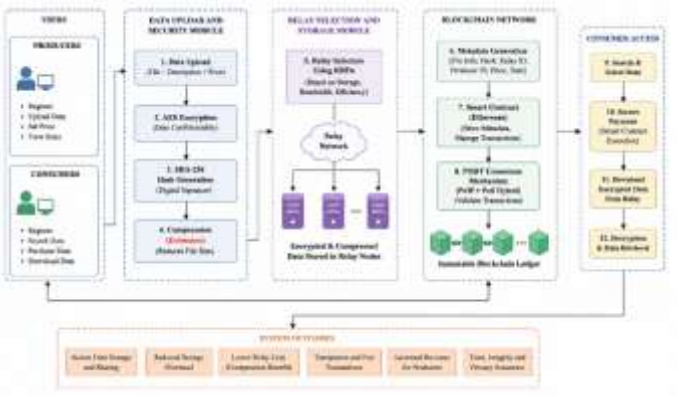

Figure 1: Blockchain-based secure data trading workflow

IV. METHODOLOGIES

Algorithm: Compression-Based Secure Data Trading Model

Input : Producer data file F
Output : Secure storage, optimized relay placement, and consumer retrieval

---

Step 1 : Start

Step 2 : Register Producer and Consumer details in Blockchain

Step 3 : Producer login authentication
If authentication fails
Terminate process
Else
Continue

Step 4 : Producer uploads data file F
Read file name, size, type, price, and description

Step 5 : Apply AES Encryption
EF = AES_Encrypt(F)

Step 6 : Generate SHA-256 Digital Signature
DS = SHA256(EF)

Step 7 : Apply Compression Extension
CF = Compress(EF)

Step 8 : Calculate compressed storage size
CS = Size(CF)

Step 9 : Execute RDPA Relay Selection
For each Relay Node Ri
Check storage capacity
Check bandwidth efficiency
Compute relay cost
End For

Step 10 : Select optimal relay node Rbest
Store compressed encrypted file CF in Rbest

Step 11 : Store metadata in Blockchain
Metadata = {File Name, Producer ID,
File Size, Signature,
Upload Date, Relay ID}

Step 12 : Execute PODT Consensus Validation
Validate transaction and create new block

Step 13 : Consumer login authentication
If valid
Allow data browsing
Else
Deny access

Step 14 : Consumer selects and purchases file

Step 15 : Verify blockchain transaction

Step 16 : Retrieve compressed encrypted data CF from selected relay node

Step 17 : Decompress retrieved data
EF = Decompress(CF)

Step 18 : Decrypt data using AES
F = AES_Decrypt(EF)

Step 19 : Deliver original file to consumer

Step 20 : Generate storage cost and revenue graphs

Step 21 : Stop

Data Collection

The methodology begins with the registration process for both producers and consumers in the blockchain-based edge computing environment. Producers are users who upload and sell digital content such as images, videos, documents, and multimedia files, while consumers are users who purchase and download those data items. During registration, user details including username, email, phone number, password, and user type are collected and securely stored in the blockchain using smart contracts. This decentralized registration process eliminates dependency on centralized servers and improves transparency and trust between users.

Blockchain Network Initialization

After user registration, the blockchain environment is initialized using Ethereum smart contracts. The smart contract contains functions for managing user details, uploaded data information, transaction records, and purchase history. Blockchain nodes maintain distributed copies of transaction records to ensure immutability and tamper resistance. Each transaction generated in the system is validated and permanently recorded within the blockchain ledger.

Producer Authentication

Once registration is completed, producers log in to the system using their credentials. Authentication is verified through blockchain-stored information to ensure that only authorized users can upload and manage data files. Successful authentication grants access to producer modules including file upload, transaction monitoring, and sales visualization.

Data Upload and Metadata Generation

The producer uploads digital files along with descriptions, pricing information, and file-related details. The system automatically extracts metadata such as file type, upload date, and storage size. This metadata is later stored in blockchain records for secure identification and retrieval of uploaded data items. The methodology ensures that uploaded files are uniquely associated with the producer account.

AES-Based Data Encryption

Before storing uploaded files, the methodology applies Advanced Encryption Standard (AES) encryption to protect confidentiality and prevent unauthorized access. Encryption converts plain data into cipher text using a secret cryptographic key. Even if attackers access relay storage, the original content cannot be viewed without proper decryption keys. This step significantly improves data privacy and security during transmission and storage.

SHA-256 Digital Signature Generation

After encryption, the SHA-256 hashing algorithm is applied to generate a digital signature for each uploaded file. The generated hash value uniquely represents file content and is used for integrity verification. If any modification occurs in stored data, the hash value changes immediately, enabling tamper detection. This mechanism ensures secure and trustworthy blockchain transactions.

Relay Node Selection Using RDPA

The methodology employs the Rounding-Based Data Placement Algorithm (RDPA) to identify efficient relay nodes for storing encrypted data files. The algorithm analyzes relay characteristics such as storage availability, bandwidth capacity, and processing efficiency before selecting suitable relay nodes. This optimized placement process reduces unnecessary relay costs and improves system performance in distributed edge environments.

Compression Mechanism

To further improve storage efficiency, the extension methodology integrates a compression algorithm before storing encrypted files in relay nodes. Compression reduces file size and minimizes storage consumption within relay systems. Smaller files occupy less relay space, decrease operational cost, and improve long-term relay utilization. This extension significantly enhances storage optimization when compared with normal blockchain storage methods.

Relay Storage and Blockchain Metadata Storage

After compression, encrypted files are stored inside selected relay nodes instead of directly saving complete files in the blockchain. Only metadata including file name, storage location, digital signature, upload date, producer details, and relay information are recorded within blockchain blocks. This approach reduces blockchain overhead, increases scalability, and improves transaction speed for large multimedia files.

PODT Consensus Mechanism

The methodology uses the Proof-of-Data-Trading (PODT) consensus mechanism for secure transaction validation. PODT combines the strengths of Proof-of-Work and Proof-of-Stake mechanisms to achieve stable, energy-efficient, and secure blockchain operations. The consensus mechanism validates uploaded data records, transaction requests, and user activities while minimizing computational energy consumption within edge computing environments.

Consumer Purchase and Secure Download

Consumers log in to the system and browse available data items uploaded by producers. After selecting a required file, consumers perform secure blockchain-based transactions to purchase the data. The system validates the transaction through smart contracts and grants authorized download access. Encrypted files are retrieved from relay nodes and decrypted securely before being delivered to consumers in original format.

Performance Analysis and Graph Generation

The final methodology stage evaluates system performance using graphical analysis and transaction monitoring. The system compares normal storage and compressed storage methods based on storage size and operational cost. Revenue graphs are generated to analyze producer income under different relay storage conditions. Experimental analysis demonstrates that the extension compression mechanism reduces storage overhead, improves relay utilization, and increases overall efficiency in blockchain-based data trading environments.

VI RESULTS & DISCUSSION

Parameters	Blockchain Storage	Compression-Based Blockchain Storage
Storage Method	Plain encrypted storage	Compressed encrypted storage
Storage Size	820 KB	410 KB
Storage Reduction	Low	50% reduction
Relay Utilization	Normal	High efficiency
Producer Revenue Ratio	0.15	0.20
Revenue Improvement	Limited	Increased by 8%
Transaction Security	Supported	Supported with AES and SHA-256
Consensus Mechanism	Traditional blockchain validation	PODT hybrid consensus
Relay Cost	High due to large storage	Reduced due to compression
Data Integrity	Basic verification	Strong hash-based verification
Blockchain Overhead	Higher metadata load	Reduced metadata overhead
File Retrieval	Secure	Secure and optimized
Processing Speed	Moderate	Faster relay access
Scalability	Limited for large files	Improved scalability
Consumer Download	Supported	Supported with secure decryption

Experimental results demonstrate that the proposed blockchain-based secure data trading framework with compression extension achieves improved storage efficiency, secure transaction management, and better producer revenue when compared with normal relay storage methods. During implementation, encrypted producer files were uploaded into relay nodes, while metadata such as file name, digital signature, upload date, and relay information were successfully stored in the blockchain through Ethereum smart contracts. The AES encryption process secured uploaded files before storage, and SHA-256 hashing generated unique digital signatures for integrity verification.

Performance analysis from the generated storage graph shows that the normal encrypted storage occupied approximately 820 KB of relay space, whereas the proposed compression-based storage reduced the size to nearly 410 KB. This achieved around 50% storage reduction, which directly minimized relay usage cost and improved relay availability for additional data uploads. The relay node selection using RDPA also reduced unnecessary storage allocation and improved placement efficiency based on bandwidth and storage capacity.

The income-cost graph generated from producer transactions indicates that the producer revenue ratio increased from 0.15 in basic storage mode to nearly 0.20 when using optimized relay storage. This improvement provided approximately 8% additional revenue for producers due to efficient relay utilization and reduced storage overhead. Transaction logs displayed successful blockchain validation with generated transaction hash values, block numbers, and smart contract execution records, confirming secure decentralized operation.

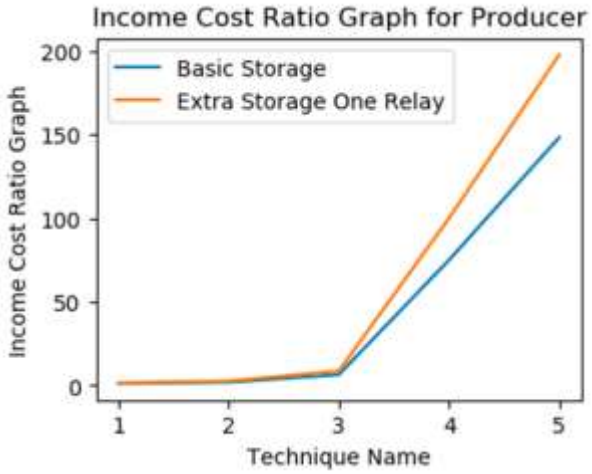

Figure 2: Income Cost Graph

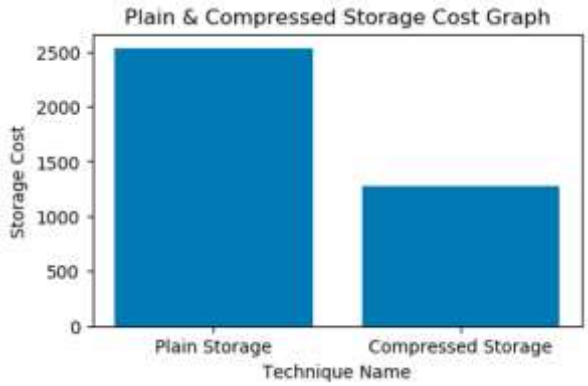

Figure 3: Storage Graph

The experimental analysis demonstrates that integrating compression techniques with blockchain-based edge computing significantly improves storage efficiency and transaction performance in decentralized data trading systems. Traditional blockchain storage methods consume higher relay space because encrypted multimedia files are stored without optimization, which increases relay cost and limits long-term storage availability. The proposed extension overcomes this limitation by compressing encrypted files before storing them in relay nodes, resulting in noticeable reduction in storage consumption and improved relay utilization.

The implementation results also show that the RDPA relay selection mechanism effectively identifies suitable relay nodes based on storage capacity and bandwidth efficiency. This optimized placement process reduces unnecessary relay expenses and improves data accessibility for consumers. AES encryption and SHA-256 hashing successfully maintained confidentiality and integrity of uploaded files, ensuring that unauthorized users cannot access or modify stored content. Smart contracts and blockchain transaction records further enhanced transparency and trust between producers and consumers.

Another important observation is the increase in producer revenue due to efficient relay management and reduced storage overhead. The generated income-cost graphs confirmed that optimized relay usage provides better profit-sharing opportunities compared with normal blockchain storage methods. The PODT consensus mechanism also contributed toward stable and energy-efficient transaction validation. Overall, the proposed framework achieved secure decentralized data trading with lower storage cost, improved scalability, and reliable consumer access in edge computing environments.

VII. CONCLUSION

The proposed blockchain-based secure data trading framework successfully improves decentralized data sharing in edge computing environments by combining blockchain technology, relay nodes, smart contracts, and compression techniques. The system eliminates dependency on centralized third-party platforms and provides secure storage, transparent transactions, and reliable data management for both producers and consumers. AES encryption and SHA-256 hashing ensured confidentiality and integrity of uploaded data, while blockchain smart contracts maintained trusted transaction verification and decentralized record management.

The integration of the compression extension significantly reduced relay storage consumption and minimized operational cost compared with normal encrypted storage methods. The RDPA relay selection algorithm improved relay utilization by selecting suitable storage nodes based on bandwidth and storage availability. Experimental analysis confirmed improvements in producer revenue, storage efficiency, scalability, and transaction reliability. The PODT consensus mechanism further enhanced system stability with lower energy consumption. Overall, the framework achieved secure, cost-effective, and scalable blockchain-based data trading suitable for modern edge computing applications.

REFERENCES

[1] X. Ma et al., “Inferring hidden IoT devices and user interactions via spatial-temporal traffic fingerprinting,” IEEE/ACM Trans. Netw., vol. 30, no. 1, pp. 394–408, Feb. 2022.

[2] “Shutterstock.” Accessed: Nov.13, 2024. [Online]. Available: https://www.shutterstock.com

[3] D. Chen and H. Zhao, “Data security and privacy protection issues in cloud computing,” in Proc. Int. Conf. Comput. Sci. Electron. Eng., 2012, pp. 647–651.

[4] J. J. Fang and K. Lei, “Blockchain for edge AI computing: A survey,” J. Appl. Sci., vol. 38, no. 1, pp. 1–21, 2020.

[5] Y. Huang, Y. Zeng, F. Ye, and Y. Yang, “Profit sharing for data producer and intermediate parties in data trading over pervasive edge computing environments,” IEEE Trans. Mobile Comput., vol. 22, no. 1, pp. 429–442, Jan. 2023.

[6] S. Nakamoto, “Bitcoin: A peer-to-peer electronic cash system,” in Proc. Decent. Bus. Rev., 2008, Art. no. 21260.

[7] D. Larimer, “Transactions as proof-of-stake,” Bitcoin Forum, Las Vegas, NV, USA, Whitepaper, Nov. 2013.

[8] D. Larimer. “Delegated proof-of-stake white paper.” 2014. [Online]. Available: http://www.bts.hk/dpos-baipishu.html

[9] M. Castro and B. Liskov, “Practical Byzantine fault tolerance and proactive recovery,” ACM Trans. Comput. Syst., vol. 20, no. 4, pp. 398–461, 2002.

[10] Y. Huang, Y. Zeng, F. Ye, and Y. Yang, “Incentive assignment in PoW and PoS hybrid blockchain in pervasive edge environments,” in Proc. IEEE/ACM 28th Int. Symp. Qual. Service (IWQoS), 2020, pp. 1–10.

[11] Y. Liu et al., "A blockchain-based decentralized, fair and authenticated information sharing scheme in zero trust Internet-of-Things," IEEE Trans. Comput., vol. 72, no. 2, pp. 501–512, Feb. 2023.

[12] X. Lin, J. Li, J. Wu, H. Liang, and W. Yang, "Making knowledge tradable in edge-AI enabled IoT: A consortium blockchain-based efficient and incentive approach," IEEE Trans. Ind. Informat., vol. 15, no. 12, pp. 6367–6378, Dec. 2019.

[13] N. Szabo, Formalizing and Securing Relationships on Public Networks, First Monday, Tel Aviv-Yafo, Israel, 1997.

[14] V. Buterin et al., "A next-generation smart contract and decentralized application platform," Ethereum, Zug, Switzerland, White Paper, 2014.

[15] H. He, A. Yan, and Z. Chen, "Survey of smart contract technology and application based on blockckain," J. Comput. Res. Develop., vol. 55, no. 11, p. 2452, 2018.

[16] A. Azaria, A. Ekblaw, T. Vieira, and A. Lippman, "MedRec: Using blockchain for medical data access and permission management," in Proc. IEEE Int. Conf. Open Big Data (OBD), 2016, pp. 25–30.

[17] T.-T. Kuo and L. Ohno-Machado, "Modelchain: Decentralized privacypreserving healthcare predictive modeling framework on private blockchain networks," 2018, arXiv:1802.01746.

[18] J.-L. de la Rosa et al., "On intellectual property in online open innovation for SME by means of blockchain and smart contracts," in Proc. 3rd Annu. World Open Innovat. Conf. (WOIC), 2016, pp. 1–13.

[19] Y. Zhang, S. Kasahara, Y. Shen, X. Jiang, and J. Wan, "Smart contractbased access control for the Internet of Things," IEEE Internet Things J., vol. 6, no. 2, pp. 1594–1605, Apr. 2019.

[20] Y. Huang, J. Zhang, J. Duan, B. Xiao, F. Ye, and Y. Yang, "Resource allocation and consensus of blockchains in pervasive edge computing environments," IEEE Trans. Mobile Comput., vol. 21, no. 9, pp. 3298–3311, Sep. 2022.

PULAPA JYOTHI SAI is currently pursuing the MCA(Master of Computer Applications) in Ideal college of Arts and science, Vidyut Nagar, Kakinada. Her research interests include Blockchain

T. PRIDHVI KRISHNA is currently serving as the Assistant Professor in the Department of Computer Science at Ideal College of Arts & Sciences(A). He possesses more than 10 years of academic and IT experience in the field of Computer Science and Engineering. His areas of interest include Software Development, Competitive Coding, Artificial Intelligence. He completed his MCA in SPACES INSTITUTE OF PG STUDIES, Afflicted by Andhra University.

He has 3 years of experienced as an Associate Consultant in Infosys Ltd.

Throughout his career, he has held various academic roles including Associate Professor, Project Coordinator, Coding Trainer.

Dr. V. S. V. Deepak is currently serving as the Head of the Department of Computer Science at Ideal College of Arts & Sciences (A). He possesses more than 18 years of academic and administrative experience in the field of Computer Science and Engineering. His areas of interest include Medical Image Processing, Cyber Security, Artificial Intelligence, Software Testing and Networking. He completed his Ph.D. research in Medical Image Processing from Swami Vivekananda University.

He has actively contributed to curriculum development, academic planning, and student mentoring. He has served as Chairman of the Board of Studies (BOS) for BCA, B.Sc. (Computer Science), B.Sc. (Artificial Intelligence), and MCA programs.