SJER Scientific Journal of Engineering Research
futuristech PT. Teknologi Futuristik Indonesia
Date of publication June 24, 2025, date of current version June 24, 2025.
e-ISSN: 3109-1725
Digital Object Identifier 10.64539/sjer.v1i3.2025.30

Article

A Thirdweb-Based Smart Contract Framework for Secure Sharing of Human Genetic Data on the Ethereum Blockchain

Tri Stiyo Famuji[1,*], Bernadine Grancho[2], Galih Pramuja Inggam Fanani[3], Hidear Talirongan[4], Raden Bagus Bambang Sumantri[1]

[1] Department of Informatics, Universitas Al-Irsyad Cilacap, Cilacap, Indonesia; tristiyofamuji@gmail.com, bagusbambang@universitasalirsyad.ac.id
[2] Marinhealth Medical Center, Greenbrea, CA, USA
[3] Department of Information Systems and Technology, Universitas 'Aisyiyah Surakarta, Surakarta, Indonesia; galihfanani@aiska-university.ac.id
[4] College of Computer Studies, Misamis University, Ozamiz City, Misamis Occidental, Philippines; hidear@mu.edu.ph
* Correspondence

The author(s) received no financial support for the research, authorship, and/or publication of this article.

Abstract: Human genetic data, crucial for advancing personalized medicine, requires secure and privacy-preserving management solutions. Traditional approaches face challenges in scalability, security, and decentralized access control. This study proposes a blockchain-based framework leveraging Thirdweb and Ethereum smart contracts to address these issues. The framework integrates decentralized storage via IPFS for cost-efficient off-chain genetic data storage, while on-chain smart contracts manage access control, encryption, and audit trails. Utilizing Solidity for smart contract development, the system ensures role-based permissions, wallet-based authentication, and immutable transaction logging. Genetic data in FASTA format, sourced from NCBI, is encrypted and linked to IPFS hashes stored on the blockchain. The architecture supports dual interfaces—command-line for developers and a Thirdweb dashboard for end-users—enabling secure data upload, access, and monitoring. Testing demonstrated functional efficacy in data integrity, access verification, and audit capabilities. Results highlight the system's ability to enhance privacy, eliminate intermediaries, and provide transparent data governance. The integration of Thirdweb further decentralizes operations, aligning with Web 3.0 principles. Key contributions include a scalable model for genetic data sharing, a customizable smart contract template, and a user-centric design. Future work should explore advanced encryption, real-world healthcare integration, and performance optimization under high-throughput conditions. This research bridges biotechnology and blockchain, offering a robust foundation for secure genomic data ecosystems.

Keywords: Blockchain; Smart contracts; Thirdweb; Genetic data; Decentralized storage; Ethereum; IPFS

1. Introduction

Human genetic data refers to the genetic information encoded in DNA, including nucleotide sequences, genes, genetic variations, and the complete genome [1]. This data serves as the foundation for understanding human genetics and has significant implications in the development of personalized medicine and targeted therapies [2]. However, due to its highly sensitive nature, genetic data must be managed with the utmost care to ensure security, privacy, and restricted access [3]. In this context, blockchain technology has emerged as an innovative solution. Blockchain is a decentralized digital ledger that enables transparency, security, and immutability in data storage [4]. Smart Contracts, which execute automatically based on predefined conditions, have been applied in various domains, including the storage of genetic data [5]. Despite the promising potential of blockchain and Smart Contracts for genetic data management, several challenges remain [6]. First, the rapidly increasing scale of human genetic data necessitates highly efficient and scalable storage infrastructure [7]. Second, privacy and security concerns are paramount, given the sensitivity of genetic information

[8]. Third, secure and verifiable data access and exchange among diverse stakeholders—such as research institutions, hospitals, and biotech companies—remain unresolved issues [9].

To address these challenges, the implementation of Thirdweb, a further evolution of the Web 3.0 paradigm, offers significant potential [10]. Thirdweb introduces a higher level of decentralization, enabling data and applications to function without reliance on intermediaries while enhancing security [11]. Thirdweb facilitates the development and deployment of Smart Contracts aligned with core principles such as decentralization, privacy, and the elimination of third-party control [12]. This enables Smart Contracts to operate within a more decentralized ecosystem, providing greater autonomy to users and data owners [13].

Through the integration of Thirdweb, this research seeks to develop innovative and secure solutions for the storage and exchange of human genetic data, while maintaining a high standard of privacy and data protection [14]. The implementation of Thirdweb in genetic data management allows for the creation of customized Smart Contracts designed for secure data sharing scenarios—such as identity-based access control, encrypted storage, and immutable data usage audit logs [15].

This research aims to contribute to the advancement of human genetic understanding and its application in more personalized and effective healthcare. Specifically, this study investigates the application of Thirdweb for developing Smart Contracts on blockchain technology to enhance the management of human genetic data, focusing on security, privacy, and scalability. By integrating blockchain via Thirdweb into genetic data management systems, this study positions itself at the intersection of biotechnology, cybersecurity, and digital innovation—proposing a novel approach to long-standing and unresolved challenges.

2. Research Methodology

This section outlines the general concept of blockchain technology, including its operational mechanisms, architecture, Ethereum platform, and smart contracts used in the development of the proposed system framework [16]. The research adopts a survey and observation strategy, based on the development of prior related studies. Figure 1 presents the sequential steps of the research methodology.

As shown in Figure 1, the initial step of this research involves identifying the core problems that must be addressed for the successful implementation of Thirdweb and smart contracts on blockchain, with the objective of ensuring secure and efficient storage of human genetic data while maintaining privacy, security, and ethical standards.

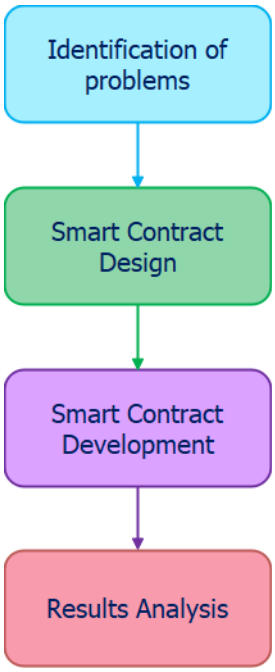

Figure 1. Research methodology.

The system design analysis focuses on evaluating the suitability of blockchain technology for storing human genetic data. This involves a comprehensive assessment of data storage requirements, privacy protocols, and security needs. Additionally, it includes an examination of blockchain infrastructure options, the development of smart contracts, and user access control mechanisms to ensure the ethical and secure management of sensitive genetic information.

The development of a blockchain-based system is the process of designing, constructing, and implementing a system utilizing blockchain technology. Such a system offers several advantages, including data security, transparency, reliability, and the elimination of intermediaries in business processes.

System testing and result analysis are critical stages in evaluating the design of the system for leveraging blockchain technology in the storage of genetic data. During this phase, the implemented blockchain-based system undergoes various testing procedures, including functionality testing, security testing, performance testing, and data integrity verification.

2.1. Related Works

Decentralized ledger technology, particularly blockchain, can be represented across multiple layers, including infrastructure, data, network, and application layers. A layered view of blockchain is presented in Figure 2, and this structure is used to illustrate the various layers and their corresponding components. To describe these layers and components, a technology-agnostic terminology has been adopted, applicable across various platforms such as Ethereum, Hyperledger, and Multichain [17].

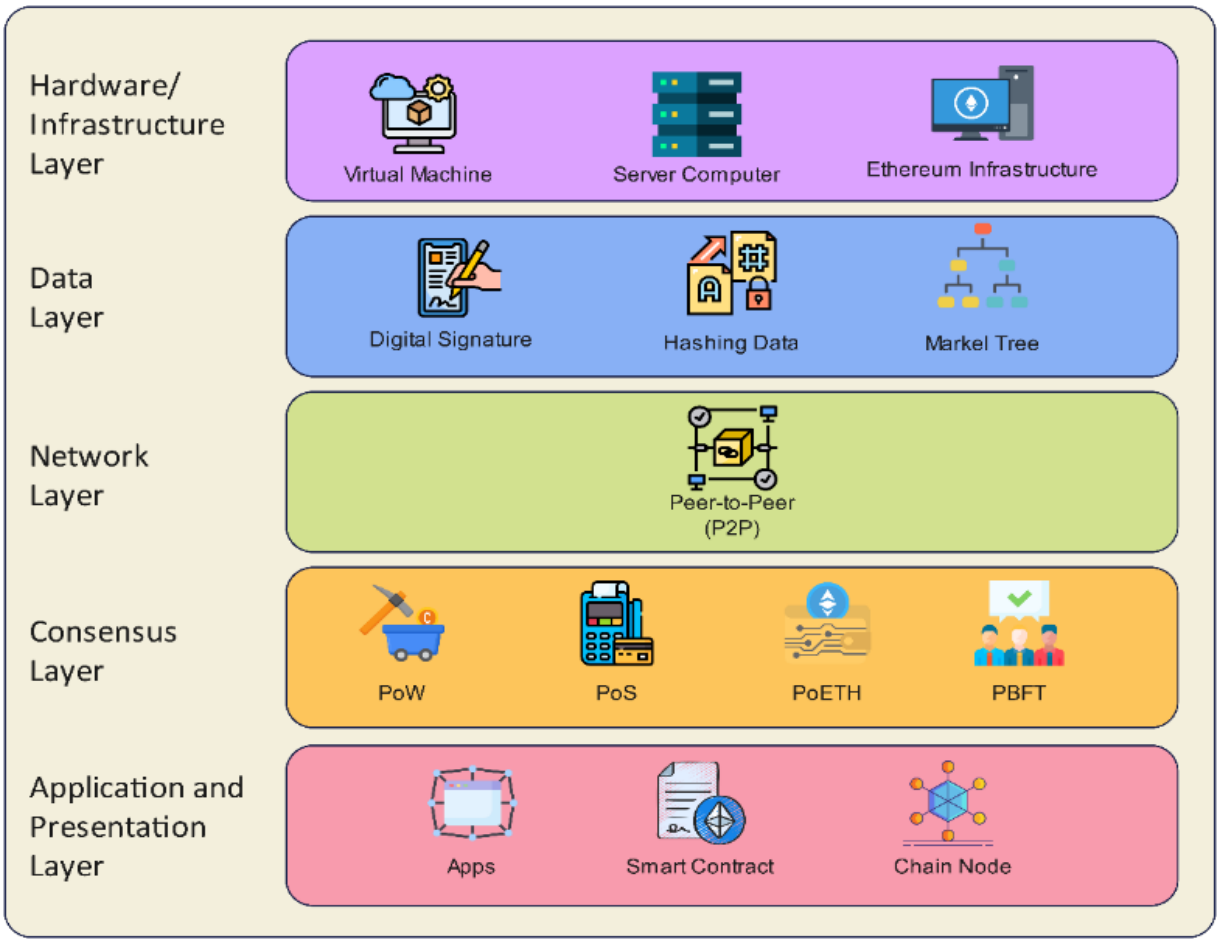

Figure 2. Blockchain Layered Architecture Model.

Based on Figure 2, the layer structure of Blockchain helps to better understand the detailed structure and functionality of blockchain, and each layer contributes to the overall success of the blockchain system [18]. The hardware/infrastructure layer is the physical layer that includes the servers where the blockchain data is hosted. The use of applications and web browsers connects to these servers through a client-server architecture [19]. However, nowadays, clients can also connect with other clients and share data through a P2P (Peer-to-Peer) network. Blockchain is an example of a P2P network that computes, validates, and stores transactions in an ordered manner within a shared ledger [20]. Each computer in this P2P network is called a "node" and is responsible for validating transactions, organizing them into blocks, and distributing them across the blockchain network.

Ethereum Infrastructure Layer: In Ethereum, there are nodes that anyone can run and participate in the Ethereum network [21]. These nodes can run "clients" such as Geth, Parity, or Pantheon to connect to the Ethereum network [22]. The Ethereum Virtual Machine (EVM) is a virtual machine that runs smart contract code and functions as a sandbox for executing that code [23]. Hyperledger Fabric Infrastructure Layer: Hyperledger Fabric, which is a blockchain based on the Hyperledger framework, consists of peer nodes that host the ledger and chaincode (smart contracts). This blockchain is typically used in enterprise environments and allows various organizations to participate in a consortium blockchain network. Peer nodes host the ledger and chaincode, and applications as well as administrators interact with these peers through the Hyperledger Fabric SDK (Software Development Kit) [24]. Data Layer: This is where all transactions and relevant information are recorded in the form of a blockchain. It is the place where all data and transactions are stored within the blockchain. Network Layer, also known as the P2P layer, is responsible for communication between nodes in the network [25]. This includes processes such as transaction propagation, block propagation, and maintaining consistency and synchronization among nodes. Consensus Layer: This layer is the core of all blockchain platforms. It includes consensus protocols that ensure all nodes in the network agree on the validity of transactions and blocks [26]. It also maintains decentralization and ensures that no single entity controls the entire network. Application Layer: The application layer includes smart contracts, chaincode, and distributed applications (dApps) that interact with the blockchain [27]. This is where business logic is executed, transactions are processed, and user interactions happen through the application interface. The structure and function of each blockchain layer is illustrated in Figure 3.

IPFS is applied to off-chain documents stored in a decentralized method. Because documents related to testing, identification, and the development of human genetic data will be expensive to store on-chain, these documents must be stored using a decentralized and secure method. Decentralized storage is an approach in information technology where data is not stored in a single data center or server, but is distributed across various locations or nodes throughout the network. Storage on IPFS is distributed and published to everyone. Therefore, the information stored on IPFS must be encrypted, and only authorized entities are permitted to read the plaintext content. For illustration of off-chain documents in figure 4.

The illustration from figure 4 depicts how IPFS (Inter-Planetary File System) is used to store documents off-chain in a decentralized storage system. Documents that are not stored directly on the blockchain, due to cost efficiency and capacity reasons, are redirected to IPFS as an alternative storage solution. IPFS works by distributing documents to various nodes in the network, rather than on a single central server, thereby forming a decentralized storage system. Each stored document will be converted into a unique hash, which can then be used to access it again. In the context of security, the document can be encrypted first before being uploaded to IPFS, so that only authorized parties can access the original information. This approach allows for the integration of storage efficiency and data security in the implementation of blockchain technology.

2.2. Blockchain

Blockchain is a database that technologically consists of a chain of digital blocks, with each block being one of several sequential links that form a digital chain. Blockchain is believed to have great potential to facilitate business processes when combined with innovative contract technology. Blockchain technology can be applied in environments with many services, such as enterprise systems [28].

Figure 5 illustrates the conceptual integration between blockchain technology, smart contracts, and enterprise systems in supporting secure and transparent digital services. On the left side, the blockchain structure is visualized as a series of interconnected blocks, starting from the Genesis Block and followed by Block 2 and Block 3. Each block contains unique data as well as a reference to the hash of the previous block, ensuring data integrity and immutability through cryptographic mechanisms. The smart contract component connected to Block 2 represents an automated digital agreement that executes instructions based on certain conditions without requiring a third party. This smart contract serves as a bridge between the blockchain and the enterprise system, which manages the institution's internal services. The enterprise system is connected to cloud infrastructure to securely store, verify, and issue digital data such as diplomas or user identities. This illustration also emphasizes a user-centered approach, involving students and administrators, with support from a distributed and protected data environment. Overall, this illustration shows how blockchain-based enterprise systems can simplify operational processes, enhance trust, and maintain information confidentiality within a multisector digital services ecosystem.

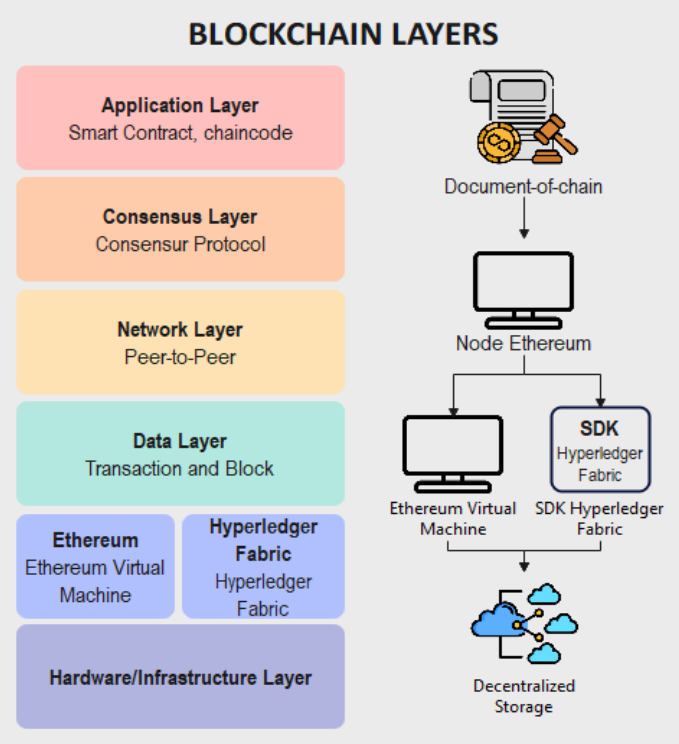

Figure 3. Structure and function of each blockchain layer.

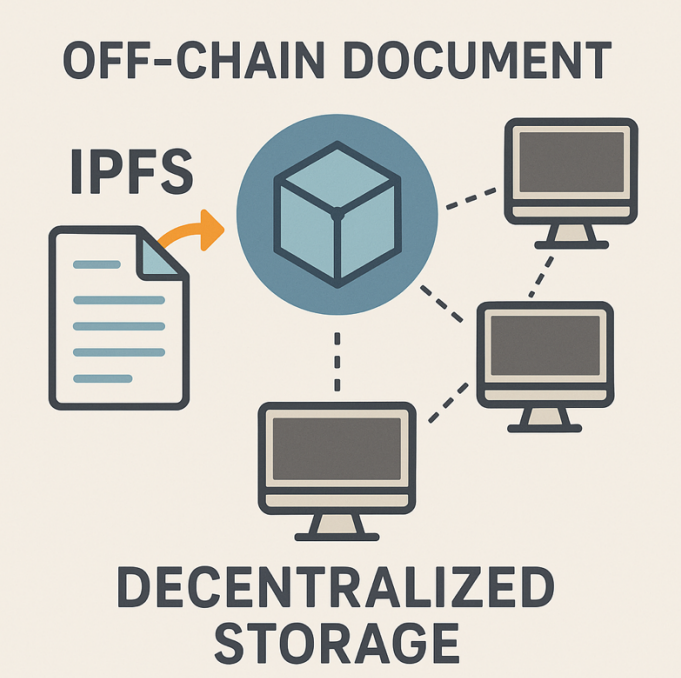

Figure 4. Off-chain documents on IPFS.

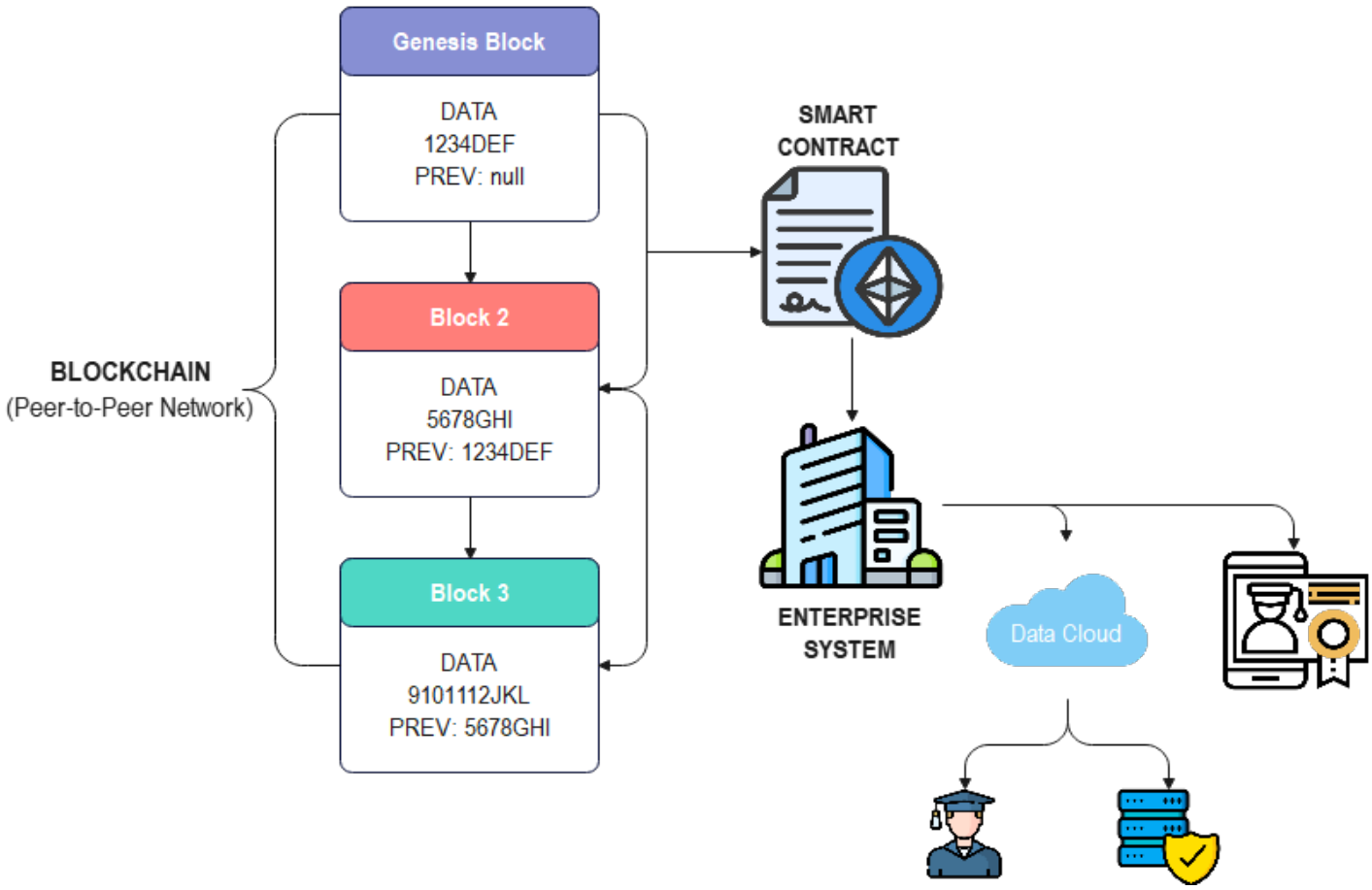

Figure 5. Conceptual integration between blockchain technology.

2.3. Ethereum

Ethereum is a network that implements distributed blockchain technology utilizing smart contracts and is open-source and programmable. Ethereum uses a cryptocurrency called Ether (ETH) as fuel to run transactions and smart contracts within its ecosystem. Ethereum has developed it and added smart contract technology, allowing blockchain technology to be used in more industries. Digital currency is an alternative form of liquidity with significant differences in ownership, transaction, and production issues compared to traditional monetary assets [29].

2.4. Smart Contract

Smart contracts are an important component of blockchain technology that enable the automation of agreement execution without the need for a third party. This concept was first introduced by Nick Szabo in 1994, and has since rapidly developed alongside the adoption of blockchain technology in various sectors [30]. Smart contract is a computer program that runs on top of the blockchain and automatically executes contract provisions when certain conditions are met [31]. In the context of blockchain, smart contracts leverage the properties of decentralization, transparency, and resistance to manipulation to ensure the integrity and reliability of contract execution. Every transaction conducted through a smart contract is permanently recorded on the blockchain, thereby minimizing the risk of fraud and increasing trust among the parties involved [32].

Figure 6 illustrates the comparison between traditional physical contracts and blockchain-based smart contracts. On the left side, the traditional process is depicted as an interaction between two parties—such as Alice and Bob—that requires physical documents, the involvement of a third party, and conventional recording systems [33]. This process tends to be slow, complex, and requires high operational costs. On the other hand, on the right side of the image, smart contracts are displayed as a digital software-based solution running on a distributed ledger (blockchain). Smart contracts enable the automatic, immutable, and verifiable recording of transactions and contracts without the need for physical intermediaries [34]. Parties such as banks, insurance companies, capital markets, regulators, and auditors still have oversight access to the system, but administrative processes become more efficient due to automation and increased transparency [35].

Overall, the integration between traditional contracts and blockchain technology results in faster, simpler, and more cost- and time-efficient processes. This supports more economical financial product innovations and more accurate reporting systems, thereby facilitating the digital transformation of the global financial system.

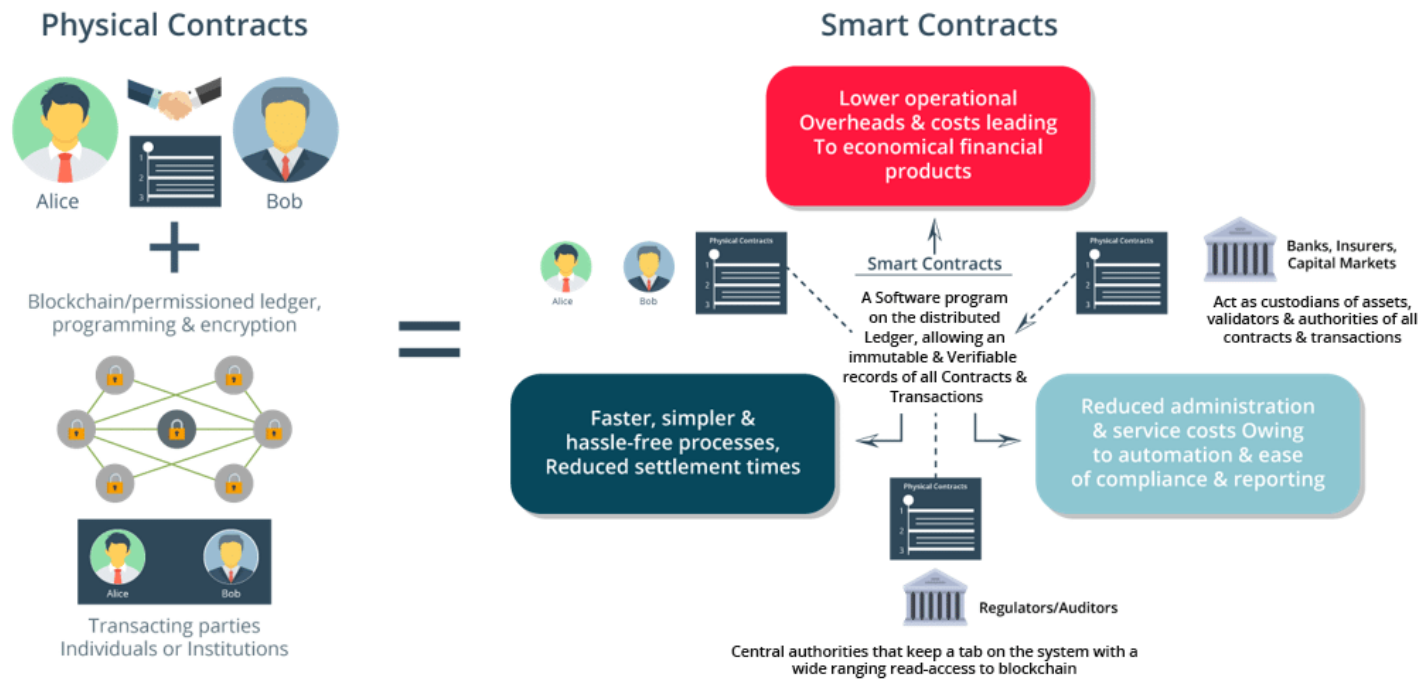

Figure 6. Comparison between traditional physical contracts and smart contract.

3. Discussion and Analysis of Results

3.1. Data Acquisition

In this study, the data used was obtained from the National Center for Biotechnology Information (NCBI) website. Where the data was obtained in Fasta format. Fasta is one of the most widely used formats for storing DNA and protein sequences. The format is a text-based format for representing nucleotide sequences or amino acid sequences. Nucleotides or amino acids are represented using single-letter codes. The main advantage of the Fasta format is the ability to use a broader scoring matrix and the facility to combine annotations into alignments.

Figure 7. Positive strand nucleotide sequence displayed in genomic analysis software.

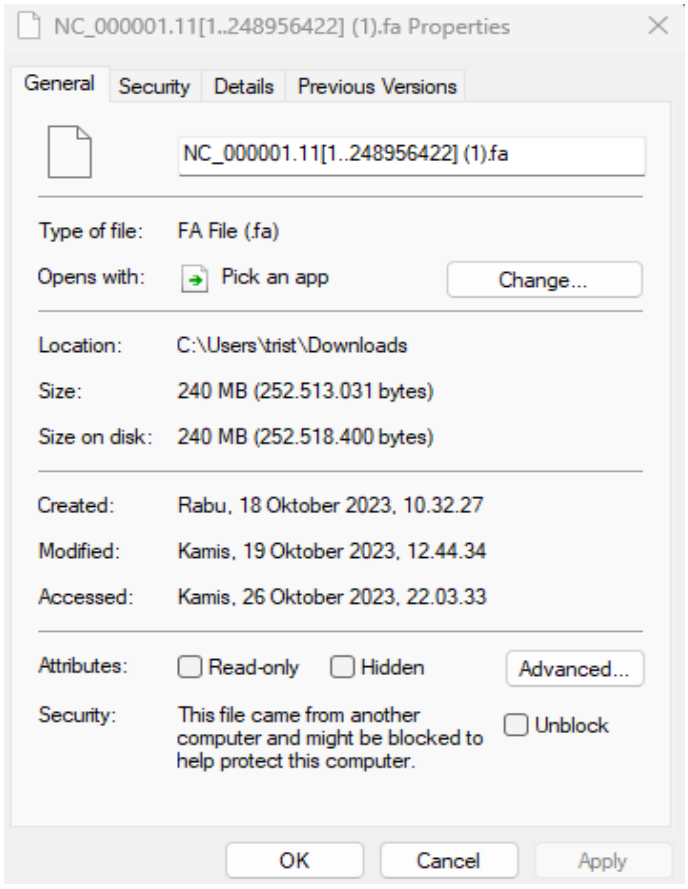

Figure 8. FASTA File Metadata of Human Chromosome.

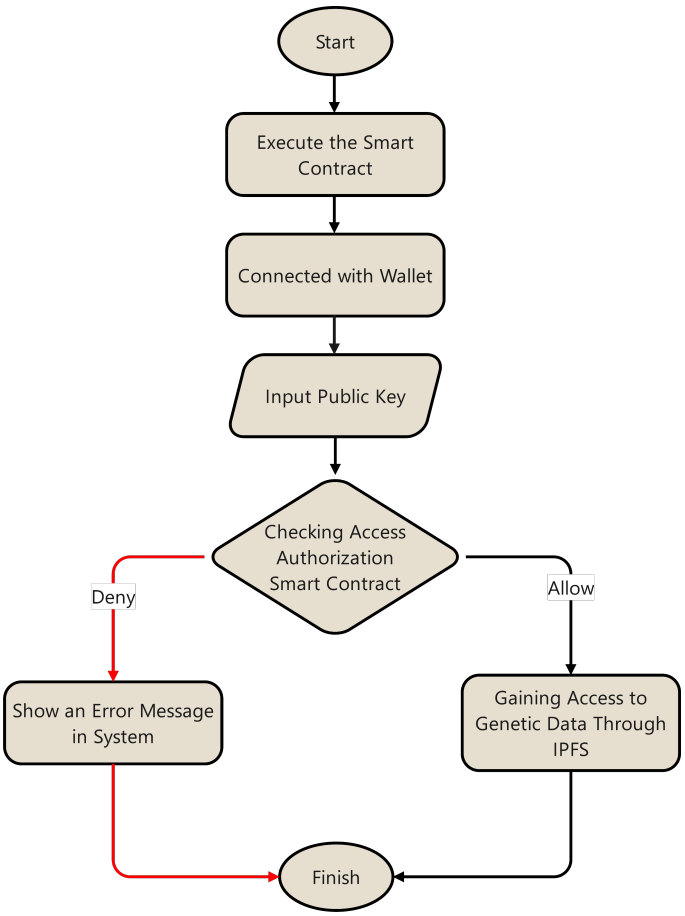

Figure 9. Flowchart of the smart contract.

Based on Figure 7, human genetic data was obtained from the sequence viewer menu on the NCBI website. Sequence view is one of the features on the NCBI website that allows users to view and examine genetic or protein sequences in more detail [1]. The downloaded data is in Fasta (.fa) format with a size of 240 MB. Can be seen in Figure 3 for details of the size and format of the data that has been downloaded from the NCBI website.

Figure 8 provides detailed information on the format and size of the downloaded data. The data is human genetic data or files in Fasta format. Fasta is a text format used to store and represent nucleotide sequences (DNA and RNA) or amino acid or protein sequences in bioinformatics. In this case, the Fasta format is a very commonly used format for representing human genetic data and other biological data.

3.2. Smart Contract Design

The development of this system uses the Solidity programming language. Solidity is a programming language specifically designed for developing smart contracts on blockchain platforms like Ethereum. Solidity provides syntax similar to C++ and JavaScript programming languages, which are used to write business logic, functions, and data structures in smart contracts. Solidity allows developers to implement business rules defined in smart contracts, which can then be executed automatically by the blockchain network. Solidity supports features such as variables, functions, flow control, object-oriented programming, and much more.

Based on Figure 9, the flowchart of the smart contract starts with connecting the smart contract to the wallet. This wallet is used to verify the financing of each transaction when executing the smart contract. The cost of the smart contract using MetaMask refers to the transaction fees incurred when interacting with the smart contract on the Ethereum blockchain network. In this context, a smart contract is a computer program that runs on the blockchain and can perform various operations, such as transferring cryptocurrency assets or activating smart contract functions. After the smart contract successfully connects with the Wallet, the next step is to verify the private key stored in the smart contract with the private key from the connected wallet account. The purpose of this verification is to divide authentication between the administrator and the user. Where the administrator's task is to upload genetic data and other supporting data, while the user's task is to access the genetic data and supporting data that have been stored in the smart contract.

Table 1. Algorithm for Adding Human Genetic Data to the Blockchain.

No	Algorithm 1: Add Human Genetic Data via Smart Contract
1	**Input:** ipfsHash, idWalletPatient, namePatient, hospital
2	**If** idWalletPatient **not registered in smart contract then**
3	Call function addPatient(ipfsHash, idWalletPatient, namePatient, hospital)
4	Grant access rights to patient for genetic data
5	**End If**
6	**Else**
7	Display error: "Patient already registered"
8	**End**

Table 2. Algorithm for Downloading Human Genetic Data from Blockchain.

No	Algorithm 2: Download Human Genetic Data
1	**Input:** idWalletPatient
2	**If** idWalletPatient **is registered in smart contract then**
3	Retrieve ipfsHash associated with idWalletPatient
4	Download genetic data from IPFS using ipfsHash
5	Allow patient to view/download genetic data
6	**End If**
7	**Else**
8	Display error: "Patient not found"
9	**End**

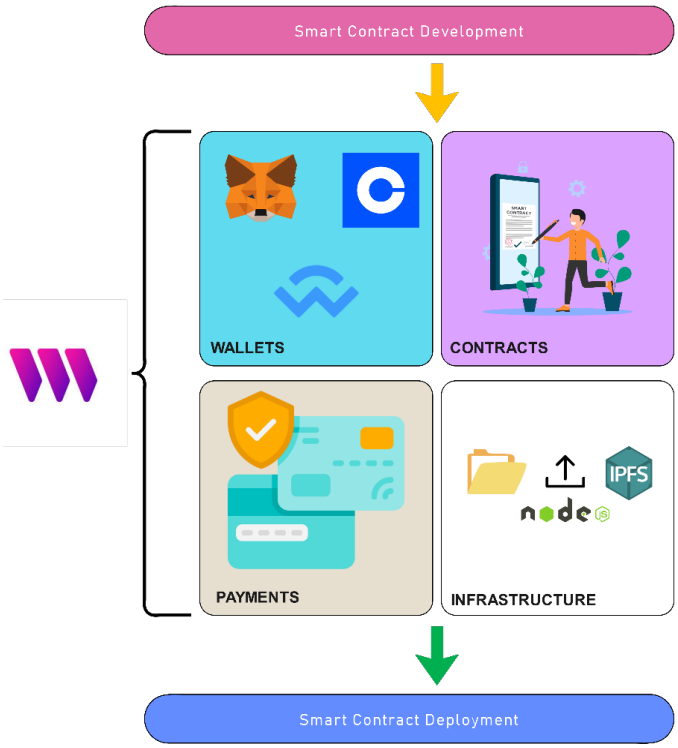

Figure 10. Component-Based Framework for Smart Contract Lifecycle Using Thirdweb.

3.3. Smart Contract Algorithm

The smart contract algorithm for storing genetic data and sharing genetic data is as follows: Algorithm 1 is the algorithm for storing patient data and patient genetic data. Algorithm 2 is an algorithm for downloading patient genetic data files based on the access key applicable to the respective patient. This algorithm also explains how to download the patient's genomic information from IPFS while ensuring that only individuals with the correct access key can obtain that information. Throughout this process, access control, encryption, and audit trails are crucial for protecting data security and privacy.

Table 1 illustrates the algorithm designed to add patient data into the blockchain-based system using a smart contract. This process begins with inputting data such as ipfsHash, idWalletPatient, namePatient, and hospital. The system then verifies whether the patient's digital wallet identity (wallet ID) has been registered in the smart contract. If not, the smart contract will permanently and immutably store the patient's data in a distributed ledger and grant the patient access rights to their data. The main objective of this algorithm is to ensure that the registration process of medical data is carried out securely, transparently, and in a decentralized manner. This implementation is crucial to prevent data duplication, enhance trust, and secure sensitive information in blockchain-based digital health service systems.

Table 2 explains the algorithm that allows the retrieval of human genetic data by patients through a smart contract. Only by using the idWalletPatient, the system will verify whether the concerned patient is registered in the smart contract. If the verification is successful, the system will search for the associated ipfsHash and use that information to download the patient's genetic data stored in IPFS. After that, the patient is granted full access to view or download the genetic information. This algorithm demonstrates how smart contracts can be used to efficiently and securely manage access control to sensitive data, without relying on third parties or centralized storage. This approach supports the principle of self-sovereign identity, where patients have full control over their own medical data.

3.4. Thirdweb Initiation

In the development of this program, Visual Studio Code is used as the IDE for programming. And using JavaScript and Solidity for smart contract implementation and integration with Thirdweb. Thirdweb is a more advanced concept in the evolution of the internet and web development. It focuses on a vision of a decentralized internet that empowers users, where users have more control over their data and services. When running a Smart Contract using Thirdweb, this involves integrating Thirdweb's principles into the design and implementation of the Smart Contract. Thirdweb aims to reduce dependence on third parties, such as large companies and centralized platforms. In the context of Smart Contracts, this means creating smart contracts that do not require intermediaries or central entities in their execution. In Thirdweb, users have more control over their personal data. This means that Smart Contracts must adhere to strong privacy principles, such as giving data owners control over access and permission for the use of their data. Thirdweb aims to eliminate central points in the network and empower users to interact directly with each other. In the context of Smart Contracts, this means that Smart Contracts must operate in a decentralized environment where network reliability is a priority. Thirdweb encourages openness and transparency in its services. The Smart Contract implemented in Thirdweb must meet these standards, allowing anyone to audit the contract and its operations. The installation of Thirdweb can be illustrated in figure 10.

3.5. Development of Smart Contract

The development of smart contracts on the Ethereum network with the implementation of Thirdweb is an innovative step in the blockchain world. By leveraging Thirdweb technology, developers can create more sophisticated smart contracts that are more connected to the decentralized internet. Thirdweb brings the concept of a decentralized web to the blockchain world, allowing smart contracts to interact with resources and data beyond the Ethereum blockchain, such as data from the conventional web. This opens up new opportunities in the development of stronger and more flexible blockchain-based

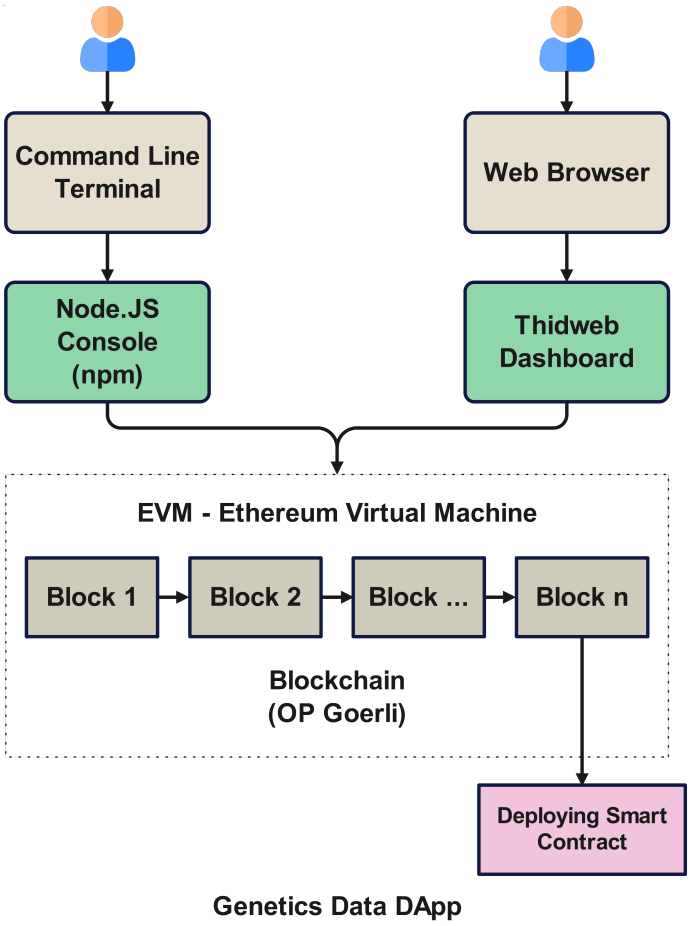

Figure 11. Genetics Data Application System Flow.

applications. With the integration of Ethereum and Thirdweb, developers can create solutions that are more transparent, secure, and reliable, enabling the development of a more advanced blockchain ecosystem connected to the outside world.

In smart contract development, the use of the Solidity programming language combined with Node.js has become a popular choice. Solidity is a programming language specifically designed for smart contracts on the Ethereum platform. With Solidity, developers can define business logic, rules, and the behavior of smart contracts. Node.js, on the other hand, is a JavaScript runtime environment often used for developing server-side applications. The integration of Node.js in the context of smart contract development allows for better interaction between client applications and smart contracts, as well as facilitating the development of web-based applications that use smart contracts as part of their backend. The combination of Solidity and Node.js helps create robust and high-performance blockchain solutions with the potential for various types of blockchain-based applications.

The proposed system architecture in this study, as shown in Figure 11, is designed to support the secure and decentralized management of genetic data by utilizing blockchain technology. This system provides two main interaction pathways for users, namely through the command line terminal and a web-based interface. The first path is used by technical users (developers) who utilize the terminal and Node.js console (npm) to run scripts, manage dependencies, and compile and deploy smart contracts to the blockchain network. The second path is intended for general users, such as researchers or healthcare practitioners, who access the system through a browser using the Thirdweb Dashboard. This interface is designed to make it easy for users to upload genetic data, verify access rights, and monitor transaction status without needing to understand the technical details of the blockchain.

Both paths are connected to the Ethereum Virtual Machine (EVM), with the OP Goerli test network used as the environment for smart contract implementation. Within the EVM, every user interaction, whether it involves data uploads or access requests, will be recorded in immutable transaction blocks. The deployed smart contract is responsible for managing role-based access control, verifying the integrity of file metadata through the Content Identifier (CID) from IPFS, and recording activity history in the form of an audit trail. With this approach, the system can ensure the security, transparency, and traceability of sensitive genetic data, while also enabling secure collaboration between institutions through an integrated blockchain network and user interface.

4. Conclusion and Suggestions

4.1. Conclusion

This research successfully designed and implemented a genetic data management system based on blockchain technology with smart contract integration using the Thirdweb platform. Genetic data obtained from NCBI in Fasta format is processed and stored using methods that ensure security, privacy, and integrity through smart contracts on the Ethereum network. By utilizing wallet verification features, encryption, and role-based access control, this system can clearly divide access rights between administrators and users. The system architecture that supports both command line terminal usage and web interface makes it easier for various groups, from developers to researchers, to access this service.

The integration of smart contracts, IPFS, and the Thirdweb dashboard enables the system to manage sensitive data in a decentralized and transparent manner. Thus, this system is capable of becoming a secure, reliable, and innovative solution for the storage and distribution of human genetic data, while also paving the way for the development of a broader blockchain-based bioinformatics ecosystem.

4.2. Suggestions

This research can be further developed by adding additional encryption algorithms, such as Advanced Encryption Standard (AES) or RSA, directly at the smart contract level to enhance the protection of highly sensitive genetic

data. Additionally, further testing of the developed system is required, particularly in terms of performance, scalability, and resilience against cyber attacks, to ensure the system's reliability under various usage conditions. The integration of this system with healthcare services such as hospitals or genomic research institutions is also highly recommended, so that the distribution and management of genetic data can be carried out efficiently and securely in the real world. From the user interface perspective, developing a more user-friendly interface for non-technical users such as researchers or medical personnel needs to be a priority, so that this system can be more widely adopted. Finally, the implementation of this system in real case studies, such as national genomic data storage projects or patient data management, will be an important step to validate the benefits and effectiveness of blockchain-based systems in the field of bioinformatics.

5. Conflicts of Interest

The authors declare no conflicts of interest.

6. References

[1] T. S. Famuji, H. Herman, and S. Sunardi, "Proses Implementasi Bioinformatika pada Digitalisasi Data Genetika Manusia," *Simetris J. Tek. Mesin, Elektro dan Ilmu Komput.*, vol. 14, no. 1, pp. 1–12, May 2023, doi: 10.24176/simet.v14i1.9064.

[2] E. V. Minikel, J. L. Painter, C. C. Dong, and M. R. Nelson, "Refining the impact of genetic evidence on clinical success," *Nature*, vol. 629, no. 8012, pp. 624–629, May 2024, doi: 10.1038/s41586-024-07316-0.

[3] C. Pereira *et al.*, "Security and Privacy in Physical–Digital Environments: Trends and Opportunities," *Futur. Internet*, vol. 17, no. 2, p. 83, Feb. 2025, doi: 10.3390/fi17020083.

[4] T. Kukman and S. Gričar, "Blockchain for Quality: Advancing Security, Efficiency, and Transparency in Financial Systems," *FinTech*, vol. 4, no. 1, p. 7, Feb. 2025, doi: 10.3390/fintech4010007.

[5] T. S. Famuji, H. Herman, and S. Sunardi, "Smart Contract Penyimpanan Data Genetika Manusia Berbiaya Murah pada Blockchain Ethereum," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 11, no. 3, pp. 695–704, Jul. 2024, doi: 10.25126/jtiik.1137558.

[6] A. Raja Santhi and P. Muthuswamy, "Pandemic, War, Natural Calamities, and Sustainability: Industry 4.0 Technologies to Overcome Traditional and Contemporary Supply Chain Challenges," *Logistics*, vol. 6, no. 4, p. 81, Nov. 2022, doi: 10.3390/logistics6040081.

[7] N. Berros, F. El Mendili, Y. Filaly, and Y. El Bouzekri El Idrissi, "Enhancing Digital Health Services with Big Data Analytics," *Big Data Cogn. Comput.*, vol. 7, no. 2, p. 64, Mar. 2023, doi: 10.3390/bdcc7020064.

[8] P. Kováč *et al.*, "Artificial Intelligence-Driven Facial Image Analysis for the Early Detection of Rare Diseases: Legal, Ethical, Forensic, and Cybersecurity Considerations," *AI*, vol. 5, no. 3, pp. 990–1010, Jun. 2024, doi: 10.3390/ai5030049.

[9] J. Merhej, H. Harb, A. Abouaissa, and L. Idoumghar, "Toward a New Era of Smart and Secure Healthcare Information Exchange Systems: Combining Blockchain and Artificial Intelligence," *Appl. Sci.*, vol. 14, no. 19, p. 8808, Sep. 2024, doi: 10.3390/app14198808.

[10] S. Rathor, M. Zhang, and T. Im, "Web 3.0 and Sustainability: Challenges and Research Opportunities," *Sustainability*, vol. 15, no. 20, p. 15126, Oct. 2023, doi: 10.3390/su152015126.

[11] A. K. Goel, R. Bakshi, and K. K. Agrawal, "Web 3.0 and Decentralized Applications," in *The 2nd International Conference on Innovative Research in Renewable Energy Technologies (IRRET 2022)*, Basel Switzerland: MDPI, Jul. 2022, p. 8. doi: 10.3390/materproc2022010008.

[12] C. Kontos, T. Panagiotakopoulos, and A. Kameas, "Applications of Blockchain and Smart Contracts to Address Challenges of Cooperative, Connected, and Automated Mobility," *Sensors*, vol. 24, no. 19, p. 6273, Sep. 2024, doi: 10.3390/s24196273.

[13] G. Palaiokrassas *et al.*, "Combining blockchains, smart contracts, and complex sensors management platform for hyper-connected smartcities: An IoT data marketplace use case," *Computers*, vol. 10, no. 10, 2021, doi: 10.3390/computers10100133.

[14] A. Musamih, K. Salah, R. Jayaraman, S. Ellahham, M. Omar, and I. Yaqoob, "Blockchain and NFT-based Solution for Genomic Data Management, Sharing, and Monetization," *IEEE Access*, 2025, doi: 10.1109/ACCESS.2025.3544643.

[15] S. Chen, Q. Cao, and Y. Cai, "Blockchain for Healthcare Games Management," *Electronics*, vol. 12, no. 14, p. 3195, Jul. 2023, doi: 10.3390/electronics12143195.

[16] G. Tripathi, M. A. Ahad, and G. Casalino, "A comprehensive review of blockchain technology: Underlying principles and historical background with future challenges," *Decis. Anal. J.*, vol. 9, p. 100344, Dec. 2023, doi: 10.1016/j.dajour.2023.100344.

[17] Y. Zhu, Q. Guo, H. Yin, K. Liang, and S. S. Yau, "Blockchain-Based Software Architecture Development for Service Requirements With Smart Contracts," *Computer (Long. Beach. Calif).*, vol. 54, no. 12, pp. 72–80, Dec. 2021, doi: 10.1109/MC.2021.3091379.

[18] A. Singh, A. Gutub, A. Nayyar, and M. K. Khan, "Redefining food safety traceability system through blockchain: findings, challenges and open issues," *Multimed. Tools Appl.*, vol. 82, no. 14, pp. 21243–21277, 2023, doi: 10.1007/s11042-022-14006-4.

[19] F. Alzhrani, K. Saeedi, and L. Zhao, "Architectural Patterns for Blockchain Systems and Application Design," *Appl. Sci.*, vol. 13, no. 20, p. 11533, Oct. 2023, doi: 10.3390/app132011533.

[20] M. Ul Hassan, M. H. Rehmani, and J. Chen, "Differential privacy in blockchain technology: A futuristic approach," *J. Parallel Distrib. Comput.*, vol. 145, pp. 50–74, Nov. 2020, doi: 10.1016/j.jpdc.2020.06.003.

[21] J. P. Eisenbarth, T. Cholez, and O. Perrin, "Ethereum's Peer-to-Peer Network Monitoring and Sybil Attack Prevention," *J. Netw. Syst. Manag.*, vol. 30, no. 4, 2022, doi: 10.1007/s10922-022-09676-2.

[22] L. Zhang and D. Kim, "A Peer-to-Peer Smart Food Delivery Platform Based on Smart Contract," *Electronics*, vol. 11, no. 12, p. 1806, Jun. 2022, doi: 10.3390/electronics11121806.

[23] Z. Wang, H. Jin, W. Dai, K.-K. R. Choo, and D. Zou, "Ethereum smart contract security research: survey and future research opportunities," *Front. Comput. Sci.*, vol. 15, no. 2, p. 152802, Apr. 2021, doi: 10.1007/s11704-020-9284-9.

[24] A. Iftekhar, X. Cui, Q. Tao, and C. Zheng, "Hyperledger Fabric Access Control System for Internet of Things Layer in Blockchain-Based Applications," *Entropy*, vol. 23, no. 8, p. 1054, Aug. 2021, doi: 10.3390/e23081054.

[25] G. Hernández-Oregón, M. E. Rivero-Angeles, J. C. Chimal-Eguía, and J. E. Coyac-Torres, "Performance Analysis of P2P Networks with Light Communication Links: The Static Managed Case," *Appl. Sci.*, vol. 13, no. 13, 2023, doi: 10.3390/app13137906.

[26] M. Pineda, D. Jabba, W. Nieto-Bernal, and A. Pérez, "Sustainable Consensus Algorithms Applied to Blockchain: A Systematic Literature Review," *Sustain.*, vol. 16, no. 23, 2024, doi: 10.3390/su162310552.

[27] M. A. Hisseine, D. Chen, and X. Yang, "The Application of Blockchain in Social Media: A Systematic Literature Review," *Appl. Sci.*, vol. 12, no. 13, p. 6567, Jun. 2022, doi: 10.3390/app12136567.

[28] I. Afrianto, A. Heryandi, and S. Atin, "Blockchain-based Trust, Transparent, Traceable Modeling on Learning Recognition System Kampus Merdeka," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 22, no. 2, pp. 339–352, Mar. 2023, doi: 10.30812/matrik.v22i2.2780.

[29] R. Afrinanda, L. Efrizoni, W. Agustin, and R. Rahmiati, "Hybrid Model for Sentiment Analysis of Bitcoin Prices using Deep Learning Algorithm," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 22, no. 2, pp. 309–324, Mar. 2023, doi: 10.30812/matrik.v22i2.2640.

[30] L. Ante, "Smart Contracts on the Blockchain – A Bibliometric Analysis and Review," *SSRN Electron. J.*, 2020, doi: 10.2139/ssrn.3576393.

[31] O. Ali, M. Ally, P. Clutterbuck, and Y. Dwivedi, "The state of play of blockchain technology in the financial services sector: A systematic literature review," *Int. J. Inf. Manage.*, vol. 54, p. 102199, Oct. 2020, doi: 10.1016/j.ijinfomgt.2020.102199.

[32] H. Taherdoost, "Smart Contracts in Blockchain Technology: A Critical Review," *Information*, vol. 14, no. 2, p. 117, Feb. 2023, doi: 10.3390/info14020117.

[33] G. Habib, S. Sharma, S. Ibrahim, I. Ahmad, S. Qureshi, and M. Ishfaq, "Blockchain Technology: Benefits, Challenges, Applications, and Integration of Blockchain Technology with Cloud Computing," *Futur. Internet*, vol. 14, no. 11, p. 341, Nov. 2022, doi: 10.3390/fi14110341.

[34] M. Bartoletti and L. Pompianu, "An Empirical analysis of smart contracts: Platforms, applications, and design patterns," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10323 LNCS, pp. 494–509, 2017, doi: 10.1007/978-3-319-70278-0_31.

[35] X. Wang, "Research on the Application of Blockchain Technology and Smart Contracts in the Financial Industry," *Front. Business, Econ. Manag.*, vol. 15, no. 2, pp. 392–395, May 2024, doi: 10.54097/gx0qhy44.