International Journal of Computer and Organization Trends
ISSN: 2249 – 2593 / https://doi.org/10.14445/22492593/IJCOT-V15I2P301

Volume 15 Issue 2, 1-14, May-Aug 2025

Original Article

Leveraging Blockchain Technology for Secure and Transparent Document Archiving

Ebiesuwa Oluwaseun[1], Alpheaus Amarachi Daniel[2]

Computer Science Department, Babcock University, Ogun State, Nigeria
Data & Archive Registry, Babcock University, Ogun State, Nigeria

[1]Corresponding Author : ebiesuwao@babcock.edu.ng

Received: 14 March 2025 Revised: 18 April 2025 Accepted: 03 May 2025 Published: 19 May 2025

Abstract - *In today's digital world, effective, transparent, and secure document management is vital for organizations in various areas. Preserving vital documents is crucial for educational institutions. Traditional document management systems frequently suffer from data security, tamper-proofing, transparency, and scalability issues. This study proposes a blockchain-based web archive management system to securely and efficiently manage student and alumni admission records, addressing the limitations of traditional paper archives. The system uses React.js for a dynamic, real-time interface and MongoDB as a NoSQL database for metadata storage. The backend uses PHP and Node.js for seamless integration with Hyperledger Fabric, enabling blockchain-based document transactions through smart contracts. PBFT consensus ensures reliable transaction validation. Blockchain integration was achieved using Express.js RESTful API. The system established robust performance, handling 100 transactions per second, 1,000 concurrent users with 300 milliseconds response time, and remaining stable with 1,500 users. It achieved high security standards, obtained positive usability feedback, and reduced operating costs to One Hundred Thousand Naira (₦100, 000) monthly. The system is designed to offer a secure, reliable, and efficient platform for managing document archives. Additionally, it is built to significantly reduce the costs connected with traditional document archive management systems. By leveraging Hyperledger Fabric, the system allows only authorized participants to access stored documents, enhancing security, data integrity, and compliance. The high throughput and low latency architecture of Hyperledger Fabric allow for the efficient handling of large volumes of documents and transactions, which is critical for the archive management system.*

Keywords *- Blockchain, Data Security, Document Archive Management, Hyperledger Fabric, Web-Based System.*

1. Introduction

The preservation of essential documents and data has long been an important part of society and academic institutions. It ensures that data is stored and preserved, temporarily or permanently, and made accessible if needed. However, as the volume of documents and data grows, more space is required for these archives. Fortunately, electronic archiving provides a fantastic solution to an age-long problem today. [1] Before the widespread adoption of web-based archiving, numerous traditional methods and technologies were implemented to archive and preserve information. Print media, microfilm and microfiche, libraries and archives, magnetic tapes, pictures and slides, and so on are some of these methods. [2] Traditional archiving methods have a long history that may be traced back to civilizations such as Mesopotamia, Egypt, and China, which created means to record and preserve knowledge on clay tablets, papyrus scrolls, and bamboo strips. [3] Traditional paper-based archiving has inherent defects that significantly disadvantage organizations, such as reduced productivity, loss of information and obstruction of information flow. Other obstacles that confront traditional archiving methods arise from the physical nature of the records involved. Physical deterioration, limited accessibility, space needs, vulnerability to disasters, limited searchability, preservation expenses, format obsolescence, security concerns, lack of redundancy, and so on are some of the major challenges. [4] As electronic transformation progresses, institutions and organizations progressively change to electronic document archiving systems. [5] There are several advantages to using web-based archiving. It is a space-saving, long-run system with nearly unlimited extension capacity. Digital archiving saves time, money, and space in terms of document management. Web-based applications use web services based on client-server architecture, request-response model, standard HTTP, and other related methods and technologies. [6] However, web-based applications face challenges like possible web failure, internet dependence, and safety and security concerns. The research gaps in the reviewed study include insufficient emphasis on security and data integrity, scalability and performance of the system under

load, user experience and training for users, interoperability and integration with existing systems, and lack of focus on regulatory compliance and cost of implementation.

By leveraging Blockchain Technology, this study seeks to develop a secure and transparent document archiving management system for academic institutions, particularly in Nigeria. The system is designed to ensure data integrity and security, operate within a decentralized framework, support interoperability and standards, enhance user accessibility and usability, optimize scalability and performance, maintain privacy and compliance, and minimize maintenance costs.

2. Related Works

2.1. Web-Based Medical Archive

The "Web-based Medical Data Archive System" proposed in [7] offers a distributed platform for managing, searching, processing, and annotating medical files, including text, audio, images, and video. It features multimodal search, online processing, annotation tools, and robust querying capabilities. Built on a three-tier browser-server architecture, the system supports any web browser on any device with internet access and uses the Oracle 9i database. However, its centralized application and database servers pose risks to Denial-of-Service (DoS) attacks and potential service outages, as all services rely on a single server.

2.2. Web-based Archive System

Furthermore, in their study "Web-based Archive Management and Student Guidance for Final Year Projects," [8] proposed a system that would assist graduating students and their supervisors in organizing and handling final year projects, with the work being transferred to the repository for archiving once completed. In the system, XAMPP was employed, allowing the webserver to function on any operating system. The system was designed using Macromedia Dreamweaver, Macromedia Flash, and Macromedia Photoshop. Malicious attacks are possible with the proposed prototype system. This vulnerability exists because the system allows a single access point to all project-related processes; it cannot guarantee sufficient protection for students' projects.

2.3. E-Health Record Management

The study by [9], "Decentralized Patient Centric E-Health Record Management System Using Blockchain and IPFS", proposed a solution to the security and privacy issues. Using the Ethereum blockchain, hospitals worldwide can communicate with one another. They use asymmetric and symmetric cryptography keys to secure the storage and access the records. It gives patients complete access to records. They stored records using IPFS. Blockchain-based EHR schemes have the potential inequality of healthcare resources, the huge carbon footprint of computational needs, and the potential distrust of health providers and patients that may ensue with the wider use of blockchain technology to deal with.

2.4. Blockchain Wrapper and Electronic Health Record Management

In addition, in their paper "A Novel Architecture for Tamper Proof Electronic Health Record Management System Using Blockchain Wrapper," [10] described a blockchain-based architecture tamper-proof EHR system. The system was developed by integrating the Ethereum blockchain into an existing system. HTML, CSS, and JavaScript were adopted to develop the front end, while Node.js and React were used for the back end. The system saved health-related data to the cloud. This tamper-proof EHR system prevents intruders from accessing patient records. The blockchain is a new technology specializing in creating tamper-resistant data systems. This system is challenged by inequality of healthcare resources, the carbon footprint of computational needs, and health provider's and patients' distrust of the wider use of blockchain.

2.5. Research Gaps

Following a careful study of current literature, noteworthy gaps were discovered. First, there is insufficient emphasis on the implementation of robust security measures and data integrity within systems. Furthermore, there are concerns about scalability and performance, notably how to manage large data volumes and preserve efficiency under severe loads. Another major gap is a lack of emphasis on user experience, which includes interface design, usability, and end-user training. Finally, there is a noticeable lack of specific considerations for how suggested solutions might be easily integrated with existing systems while maintaining established standards.

2.6. Overview of Blockchain

Satoshi Nakamoto invented the concept of digital currency, also known as cryptocurrency. Blockchain techniques were introduced in Bitcoin, where each block is linked to previous blocks using a hash value. Once a transaction has been made, it cannot be altered. Blockchains are gaining popularity among academics and scientists for various reasons, including access control, data protection, privacy, and wireless network decentralization. [11] Decentralized networks serve as the foundation for these coins, and blockchain technology powers them, making blockchain one of the most promising, disruptive, and revolutionary technologies in the tech space. [12] Distributed Ledger Technology (DLT) is one of the most promising inventions in the field of information technology, with the potential to transform economic, social, and industrial organization and collaboration. [13]

2.7. Structure of Blockchain

Each blockchain, as shown in Figure 2, comprises several blocks with a block header and body. The block header contains a variety of meta-data about the current block. Examples are timestamp, blockchain body hash value, and prior block hash value. The block body is typically used to record the actual data of current transactions. [14]

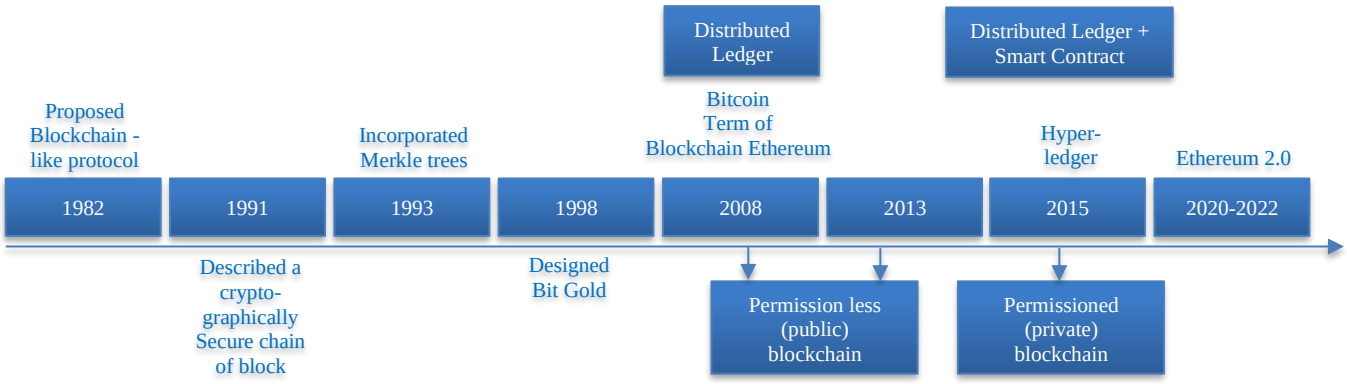

Fig. 1 Summary of the history of blockchain [14]

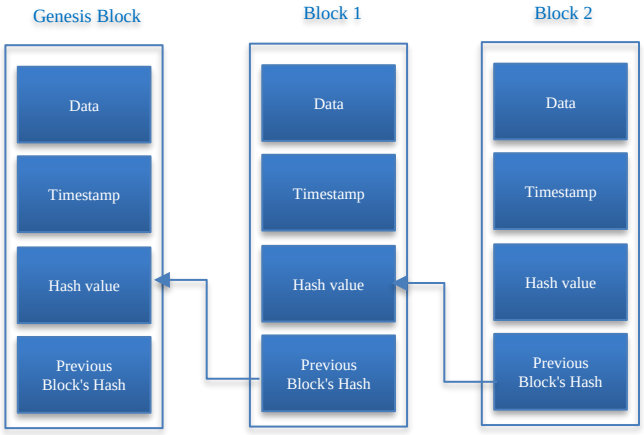

Fig. 2 Structure of blockchain [15]

2.8. Types of Blockchain

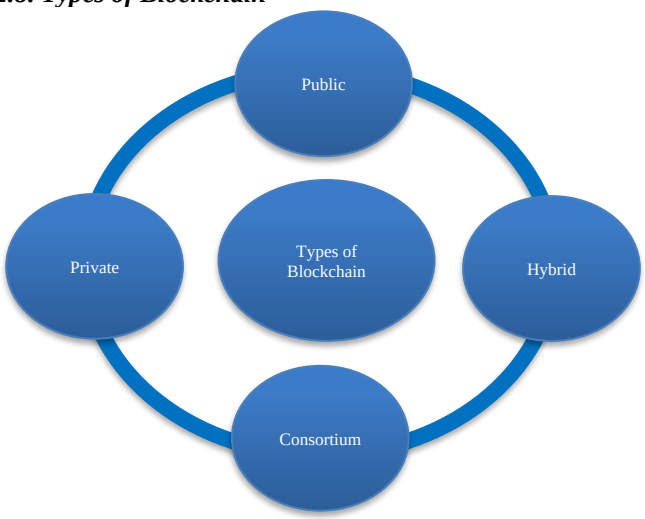

Fig. 3 Types of blockchain [17]

There are three major classifications of blockchain: public, private, and consortium, which are based on the permission/access required to join or leave the blockchain network. [16] The fourth classification is the hybrid blockchain, which combines the characteristics of both the public and private blockchains.

2.8.1. Public Blockchain

Public Blockchain is a major type of Blockchain that is open and decentralized. Computer networks are essentially accessible to anyone interested in transactions. Based on validation, the validated participant essentially receives the transaction rewards. Furthermore, two kinds of proof-of-work and proof-of-stake models are used. Furthermore, a public blockchain is a non-restrictive and distributed ledger system that does not seek any permission. This classification of Blockchain also gives authorization to verify current and past records. It is mostly secure when following strict security rules. However, non-following the security protocols may be risky. Examples of public blockchain are Bitcoin, Ethereum, and Litecoin. [18]

2.8.2. Private Blockchain

Private Blockchain is a permissioned blockchain. In a private/permissioned blockchain, the number of nodes is limited, and each node keeps a copy of the blockchain. The consensus mechanism is not very expensive for publishing a new block. Every node in the private blockchain is known, so the risk of Sybil's attack is eliminated. Hence, permissioned blockchains perform better than permissionless blockchains. Private Blockchains are most applicable in organizations where limited nodes are accessing the blockchain. [19]

2.8.3. Consortium Blockchain

The consortium blockchain, composed of a group of identified and trusted nodes, establishes supportive relationships among one or more entities and organizations. The role of the consortium blockchain is to provide participants with a secure, efficient, and trusted environment, promoting the development of data sharing and transactions. Consortium blockchains have characteristics such as multi-party participation, restricted access permissions, and high privacy protection. Furthermore, they offer advantages in traceability and auditability, enabling easy tracing of transaction roots and history. By selecting appropriate consensus techniques, consortium blockchains can provide higher performance and scalability for handling transactions and data, meeting the specific needs of organizations or industries. [20]

2.8.4. Hybrid Blockchain

Hybrid blockchain combines the elements of both public and private blockchains. By setting up a private permission system alongside a public permissionless system, an organization can control what data in the blockchain is publicly accessible and who can access it. A hybrid blockchain does not characteristically publish transactions and records but can be confirmed through smart contracts when a participant needs it. It is still possible to verify the confidentiality of the information inside the network. A private entity may own the hybrid blockchain but cannot alter transactions. The user has full network access when they join a hybrid blockchain. The user's identity is protected unless other users engage in a transaction. When this happens, the other parties' identities are revealed. [21]

2.9. Consensus Algorithm

The consensus algorithm, the blockchain's foundational technology, determines which nodes have the authority to record transactions and allows them to agree on the information included in a block quickly. This assures data consistency and security while increasing the blockchain's processing efficiency. [22] Consensus algorithms enable users or nodes to coordinate in a distributed and decentralized environment. They ensure that all entities in the system collectively agree on a single source of truth, even if entities fail. Built to ensure fault tolerance, this mechanism helps to maintain reliability in networks composed of unreliable nodes. [23] Consensus mechanism in blockchain is critical in maintaining three major properties that uphold the efficiency of the underlying network. Beyond establishing a consistent global state for a distributed ledger, it ensures the network's safety, liveliness, and fault tolerance. [24]

2.9.1. Practical Byzantine Fault Tolerance (PBFT) Consensus Mechanism

The consensus framework implemented in this system is the Practical Byzantine Fault Tolerance (PBFT) consensus mechanism. PBFT algorithm was initially proposed to solve the Byzantine General Problem and has since been applied to the blockchain consortium chains. The fault tolerance rate of this algorithm is 1/3, meaning it ensures the flexibility and security of the consensus process if it does not exceed 1/3 of the total nodes. Additionally, the PBFT algorithm significantly enhances the efficiency of the Byzantine fault-tolerant algorithm by reducing its complexity. When the number of Byzantine nodes in the system remains lower than the 1/3 threshold, the PBFT consensus algorithm ensures the system operates correctly and reliably.

This greatly improves the security performance of the blockchain system. [25] Although the PBFT algorithm significantly enhances the consensus performance of the blockchain, it faces the following challenges: a low fault tolerance. The defective nodes in the whole system cannot exceed 1/3 of the total number of nodes in the network, and the fault tolerance rate cannot be improved. Communication costs in the PBFT are quite high. Due to the unavoidable existence of faulty nodes in the network, the communication cost and processing time increase. If the total number of nodes in the system is "N", the communication complexity time is O (N2). Considering the three-phase protocol, the message transmissions can be as high as "2N (N-1)". If the number of nodes in the network exceeds a certain number, the PBFT algorithm performance will drop drastically, potentially leading to a bottleneck. Furthermore, the master node has a high probability of being malicious. This is so because the PBFT algorithm has no robust mechanism for the selection of the master node; it is likely that malicious nodes will be selected during the selection process, which will negatively affect the operational efficiency of the system. PBFT provides robust security guarantees against Byzantine faults, assuring the blockchain's integrity and consistency even in the presence of malicious nodes. [26]

2.9.2. Major Challenges of Blockchain

Scalability

Blockchain scalability concerns arise from low transaction throughput, high latency, and increasingly high resource demands. As the volume of transactions increases, the storage space needed for the Blockchain also increases.

Performance Issues

Blockchain systems suffer performance challenges, including throughput bottleneck, transaction latency, and storage limitations. For instance, smart contracts are executed sequentially by miners and validators, which significantly restrict the throughput.

Cost of Decentralization

Although decentralization is considered one of the key characteristics of Blockchain, it comes with a cost. One open issue is achieving a balance between security and resource efficiency with a consensus algorithm regarding adaptively managing the replication factor in shards.

Irreversible Bugs

Because of blockchain's immutability, fixing bugs in deployed smart contracts is significantly difficult, and there is no way to patch a flawed smart contract without reversing the blockchain, which is highly complex. Even if there is a way to update the defect, deploying a new contract version does not automatically transfer data stored from the previous version, requiring manual updates that make it quite unwieldy.

Energy Inefficient

The Proof of Work (PoW) consensus approach used in Bitcoin is highly energy-inefficient, consuming approximately 15.77-Terawatt hours of electrical energy to reach consensus, which is 0.08% of total global electricity consumption. Most of this power is spent in calculating the irreversible SHA256 hashing function. Additionally, the

Blockchain requires a large volume of time, energy, money and computing power to verify transactions, coupled with the inefficient use of scarce energy resources for financial activities, which constitutes a serious threat to the global climate due to greenhouse gas emissions. [27]

3. Methods

The design adopted in this study is highlighted below;

- The Graphical User Interface (GUI) was created using HyperText Markup Language (HTML) and Cascading Style Sheet (CSS).
- The system is powered by a blockchain engine implemented in a peer-to-peer network. A two-tabled database stores basic user information, such as username, password, school name, department name, and course. This will be used to provide first-level authentication and ensure that nodes joining the blockchain are verified. Also, a user cannot join the blockchain unless authenticated and permitted by the node that created the blockchain.
- The system was assessed by contrasting the proposed Document Archiving Management System designed leveraging blockchain technology with the existing traditional archive systems.

3.1. Web Design

JavaScript was used to build an interactive and dynamic user interface, and Hyper Text Markup Language (HTML) was used for the Graphical User Interface. Node.js SDK integrated the web application with Hyperledger Fabric blockchain networks.

User Experience: the interface is designed to be accessible and responsive on many devices, such as smartphones, desktops, and tablets. It is built with a clear and consistent layout with easy-to-understand icons, buttons and menus.

Tools Adopted

- Figma and Adobe XD were used for detailed UI/UX design and prototyping.
- InVision was adopted for interactive modeling and user feedback.
- Balsamiq was used for building low-fidelity wireframes.

3.1.1. Frontend Development

Tools Adopted

- React framework was used to build A Single-Page Application (SPAs) that increases user experience, enhances performance, and maintains data integrity.
- JavaScript was the backbone framework for creating a dynamic and responsive interaction within the application and for API calls.
- HTML5 and CSS3: For structuring and styling the web interfaces.
- API: This is used to make HTTP requests to interact with the backend. This is crucial for fetching and sending information between the frontend and backend.

React, an open-source JavaScript library, is at the core of frontend development. It ensures the application is reliable, effective, and easy to use. The library was used to create a Single-Page Application (SPA) that improves user experience. The tool's effective update approaches provide a responsive, seamless, easy-to-use interface. React improves efficiency and preserves data integrity throughout the application by re-rendering only the interface elements that require modification, ensuring that all nodes view precise and reliable information.

The backbone of the application is JavaScript. This adaptable language was used to design a dynamic and responsive interaction, resulting in a vivid and engaging user experience. JavaScript's ability to handle events and alter the Document Object Model (DOM) in real time is critical to the application's operation.Furthermore, JavaScript makes API calls, allowing the frontend to interact effectively with the backend. HTML5 and CSS3 were utilized to design and style the web pages. HTML5 provides a strong and semantic structure for web pages, making the pages accessible and clear for users and search engines. CSS3, on the other hand, improves designs by providing colorful and responsive styles. This combination ensures that the system is functional and visually beautiful, resulting in an excellent user experience across various devices. Application Programming Interface (API) is an important component in the system design. It enables nodes to send HTTP requests to the backend. This interactive channel is crucial for retrieving data from the server and sending fresh information back to the server. The API keeps the frontend and backend in continuous interaction, allowing for real-time changes while ensuring data integrity.

3.1.2. Backend Development

The backend was developed to handle business logic and integrate with the Hyperledger Fabric network.

Tools Adopted:

- Node.js/Express.js was used to build the server-side application and RESTful APIs.
- Hyperledger Fabric SDK for Node.js was used to interact with the Hyperledger Fabric blockchain network.
- MongoDB: NoSQL databases for storing document metadata on the blockchain. Utilizing MongoDB for metadata storage ensures the system achieves high performance, scalability, and flexibility while facilitating seamless integration with blockchain technology.
- Interplanetary File System (IPFS) was used to store documents off-chain and to ensure decentralization, scalability, and integrity.
- JWT (JSON Web Tokens): For secure user authentication and authorization.

Blockchain Integration: The backend was integrated with the blockchain network using the following tools:

Hyperledger Fabric: This is the blockchain platform for managing the document archive. Hyperledger fabric was set up in a local machine with Docker installed. Docker components are configured; this includes Docker engine and Docker images for Hyperledger/fabric-peer for peer nodes, Hyperledger/fabric-orderer for orderer nodes and Hyperledger/fabric-ca for fabric Certificate Authority. Additionally, Docker container - peer container, orderer container, CA container, and Docker compose were set up for orchestration.

Smart contract (Chaincode): To develop a smart contract, the environment was set up by installing Node.js and the Hyperledger Fabric SDK. This was designed to handle document activities (upload, search, retrieve, and update) while also combining the frontend and backend. Docker was used to deploy and manage the Hyperledger Fabric network and chaincode in containers. A new directory was built for the chaincode project to provide proper structure version control and avoid conflicts with other components. To ensure the right foundation for the smart contract and to guarantee that all logic is in one place and ready for deployment, the smart contract file - chaincode.js - was created in the project directory. The contract code is written in JavaScript and packed as a tar.gz containing the chain code. The chain code was made available on the blockchain so that peers may interact with it.

3.2. Integration of Hyperledger Fabric Blockchain and Document Management

Hyperledger Fabric is a suitable choice for the development of the proposed system due to key features and benefits that align with the system's requirements. Here are the primary reasons: Hyperledger Fabric, a notable framework under the Linus foundation, offers a strong basis for creating secure and scalable enterprise blockchain systems. Its modular architecture enables precise configuration of network components, ensuring privacy and confidentiality among participants in the network. This flexibility allows customization and scalability, positioning it for various document management scenarios.

Furthermore, its permissioned network model ensures that only verified and authorized entities can participate. [28] Creating the blockchain network for the system involves several crucial components and configurations to guarantee a secure, efficient, and robust architecture. Hyperledger Fabric network was established with the Data and Archive Unit of the Registry, which was designated as the primary department with a set of peers (nodes). Smart Contracts (Chaincode) were developed using Node.js to define document management's operational rules. These rules include authenticating user permissions before document submission or retrieval, logging document access events for auditing purposes, and enforcing immutability to preserve data integrity across transactions. A dedicated channel, named docarchive-channel, was configured for Registry-specific operations.

The genesis block was used to initialize this channel, embedding predefined policies for governance, access control, and data sharing among participants. Hyperledger Fabric's Certificate Authority (CA) was used to issue user's and organizations digital certificates. These certificates facilitate secure authentication and ensure only authorized entities can interact with the blockchain network.

3.3. Express.js RESTful API Implementation

Express.js is a flexible Node.js web application framework used to build the systems web application. It provided a robust environment for creating RESTful API and made handling HTTP requests and responses easy. These attributes make it a robust solution for creating a RESTful API. The framework handles the following critical operations: [29]

1. User authorization and authentication: the framework secures access to the system using a JWT token and offers permission based on assigned roles.
2. Management of document: it performs Create, Read, Update, and Delete (CRUD) operations using chain code for stored document Meta Data on the system.
3. Interaction with blockchain: it creates the enabling environment for seamless interaction with smart contracts to store document Meta Data on the blockchain network, confirm document authenticity and perform access and modify transactions on the system.
4. Performs audit trail: Express.js tracks and verifies activities on the immutable logs for document access and modifications.
5. Search and filtering: The framework allows users to retrieve documents from the system by specifying document metadata or document type.

Express.js RESTful API was implemented using the following steps:

1. Express.js application was set up by:
   - Installing dependencies such as express, cor, doteny, body-parser, jsonwebtoken, and fabric network;
   - Set up a basic server structure to handle API requests.
2. API endpoints for CRUD operations were defined to establish a seamless interaction between the backend (Hyperledger Fabric and database).
3. To ensure adequate security for the system, middleware was set up to intercept requests and enforce authorization, authentication using Json Web Token, and threat detection.
4. Express.js API and blockchain (Hyperledger Fabric) were integrated, involving smart contracts and Hyperledger Fabric SDK.

3.4. User Requirements

The system is designed to handle document uploads, searches, and downloads. It will feature user and admin interfaces, with each type of user assigned specific roles, privileges, or rights. The Administrator will be able to manage users and contacts by adding, deleting, and editing their personal details within the system. Additionally, the Administrator can upload, retrieve, and view documents.

3.4.1. Node

A node comprises every computer on a blockchain network that participates in processing transactions. These nodes connect with each other in a network and help verify transactions. They can also view all the transactions processed in a blockchain. In this application, the two primary players are:

1. End-user
2. System Administrator

3.4.2. End User

The end user performs the following tasks:

i. Login
ii. Logout
iii. View Document
iv. Search Document
v. Create stream
vi. Upload Document

This is represented pictorially in Figure 4

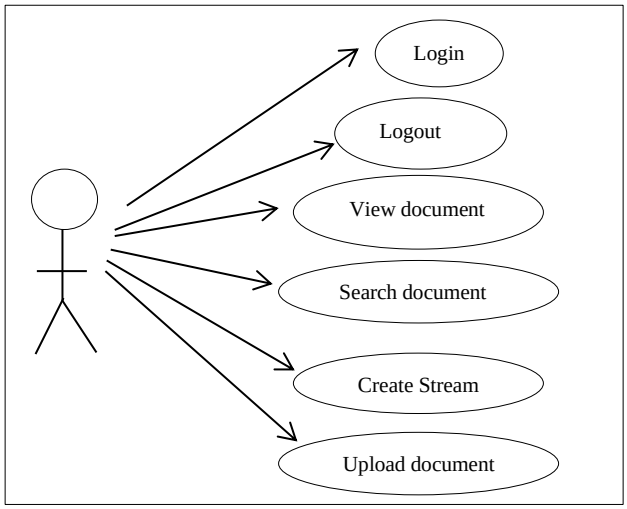

Fig. 4 Use case for end-user node

3.4.3. System Administrator

The system Administrator can carry out the following tasks:

i. Add Users
ii. Remove Users
iii. View All Documents
iv. Edit All Documents
v. Assign User
vi. Assign Roles

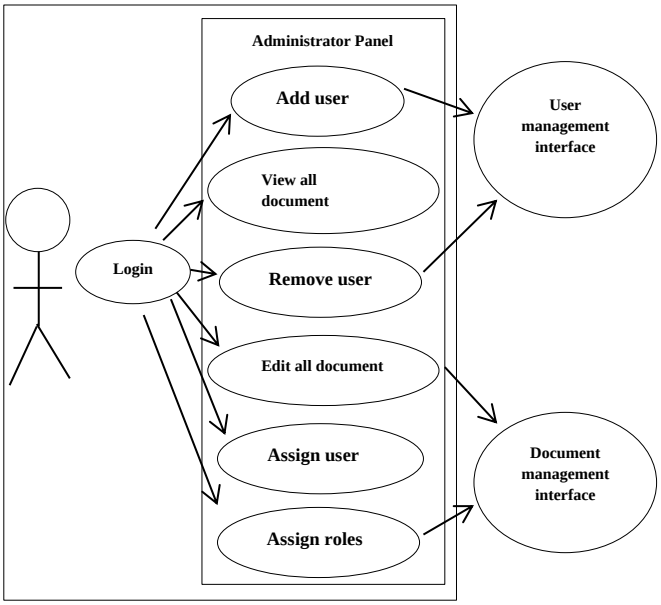

Fig. 5 Use Case for System Administrator

3.5. System Architecture

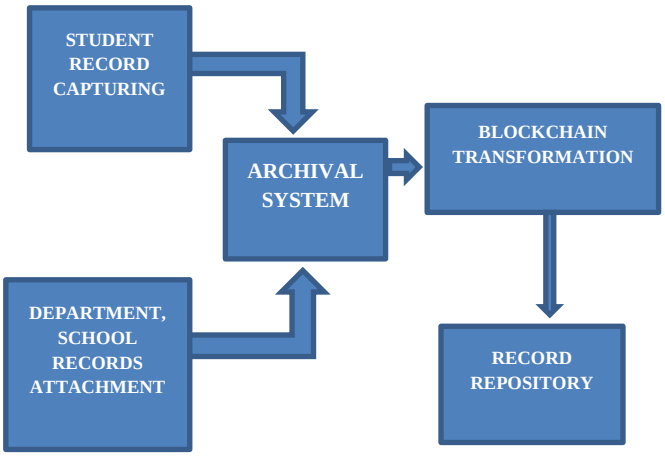

Fig. 6 Architecture of the design

3.5.1. Student Record Capturing

Transparent, impenetrable, and effective student data management in a web-based archive system is made possible by leveraging blockchain technology's decentralized and secure capabilities. This improves speed, security, and transparency while providing a dependable, auditable platform for handling student records throughout their academic careers.

3.5.2. Archival System

Digital records are managed, stored, and retrieved securely thanks to the archiving system. Key capabilities, including document submission, smart contract execution, version control, timestamping, IPFS-based decentralized storage, blockchain metadata storage, and access control enforcement, are all supported when integrated with the larger document management system.

3.5.3. Record Repository

Storing records in a blockchain-based web archive system involves digitizing physical documents, assigning metadata, generating a cryptographic hash, initiating blockchain transactions via smart contracts, storing files securely on IPFS, and encrypting documents before storage.

3.5.4. Database Design

The application includes a database component to store basic user information for first-level authentication of nodes joining the blockchain.

Application of Blockchain-Based Document Management System

The blockchain-based document management system has potential applications across different sectors:

Healthcare: Patient records, medical history, and test results can be archived in the system. Using encryption and consensus mechanisms, patients' records are secured.

Education: The system can be used to manage student and graduate records, certificates, transcripts, and admission credentials. The system can checkmate fraud and ensure authenticity through tamper-proofing of stored documents.

Financial Services: Financial records such as transaction records, audit logs, budget approvals, payment receipts, and contract records to facilitate real-time audit trails.

Real Estate Document Management: A blockchain-based document managing system can manage leases, sales agreements, and property deeds. This provides an immutable record of ownership and transactions. It enhances the ease of property transfer using smart contracts.

4. Results

The deployment of the blockchain-powered web-based document archive management system has yielded remarkable results across multiple functional and technical areas. This cutting-edge solution integrates decentralized storage, advanced encryption, and user-friendly design to tackle key challenges in document management effectively. Highlighted below in Table 1 are its main features and the outcomes achieved:

Fig. 7 Archive dashboard

4.1. Overview of the Archive Dashboard

The system dashboard provides users in various roles with an easy and efficient view that improves user accessibility and operational efficiency, as well as the ability to easily login and explore the dashboard. The dashboard is integrated with blockchain technology and includes a variety of interactive icons for easy navigation and real-time data.

4.1.1. Icons and Functionalities

- School: Provides centralized access to all documents and records within a school.
- Department: Offers a customized view for department-specific document archiving and management.
- Record Activities: Allows access to view records of securely logged activities in the blockchain.
- Documents: This icon grants access to the repository for document submissions, retrieval, and integrity verification with blockchain hashes.
- Site traffic overview: This icon opens access to analysis of real-time information on system use, user activity, and peak traffic periods.

4.1.2. Benefits

The dashboard offers a curated lens through which activities on the system are viewed at a glance. It combines visual representation and graphical embellishments to provide layers of abstraction and simplification for numerous related data points so that users get a real-time overview of the most relevant information for critical decision-making. [30] With the integration of blockchain technology, all activities on the system are well documented in immutable logs, ensuring a high level of accountability and trust. Furthermore, real-time traffic insights enable administrators to monitor system utilization better, improving efficiency and assuring scalability to manage different user activity levels.

4.2. Add New Record Interface

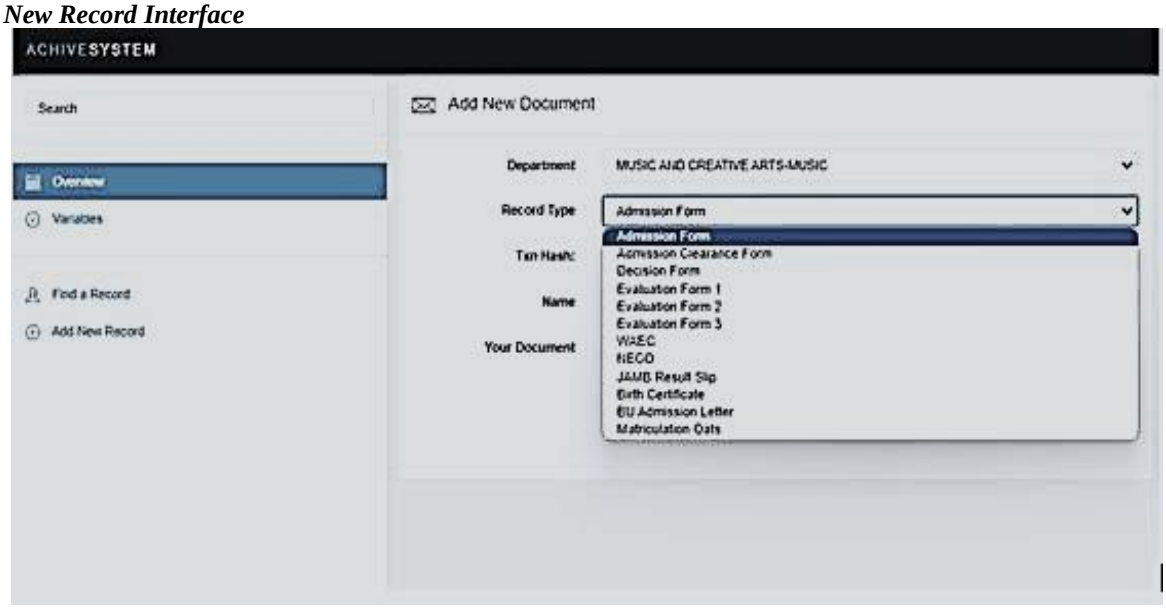

Fig. 8 Add new record interface

Adding a new record to the system is a simple and intuitive process that ensures accuracy and efficiency. Users start by selecting the relevant department from the appropriate school category. This ordered structure makes navigation easier while also allowing for accurate record classification. After selecting the department, the document type can be defined, ensuring each entry is correctly labeled and easily retrievable. This systematic approach simplifies record-keeping, promotes consistency, and makes accessibility of stored records easy for users.

4.3. Document Entry and Hash Window

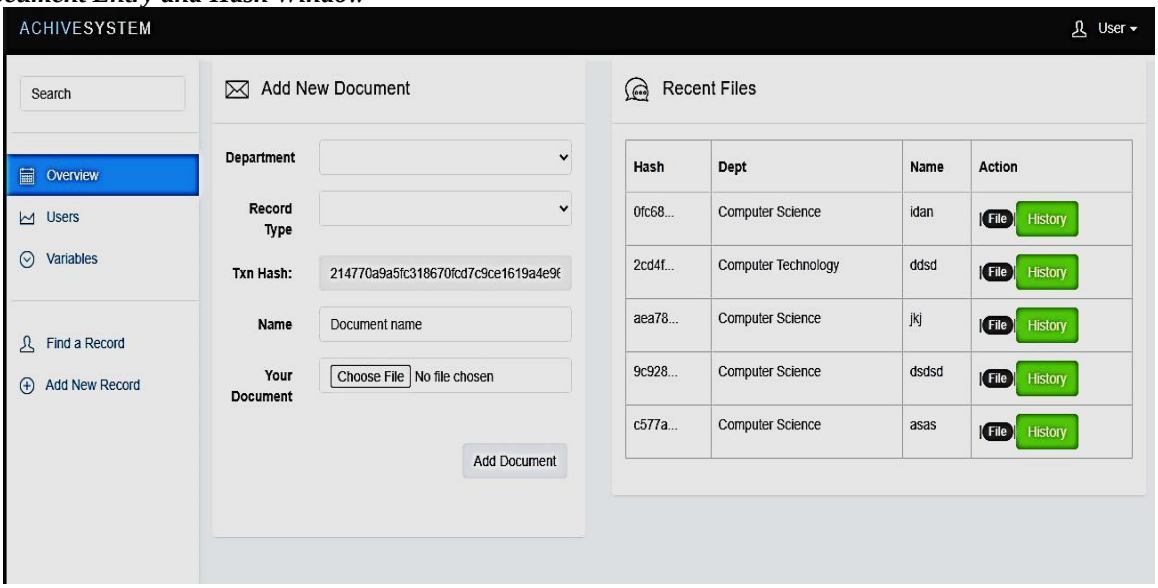

Fig. 9 Document entry and hash window

Document entry and hash window display the integration of Hyperledger fabric into the document management system to enhance security, transparency and data integrity. Adding a new record leads the system to collect all the necessary document details and submit them to the blockchain. The Meta data is organized to meet the predefined schema for the selected record type. The attached document is hashed to generate a unique hash of the document.

4.4. Implementation Results

Table 1. Implementation results

Attribute	Description	Result	Measurement Tool(s)
Security	Blockchain encryption and tamper-proof mechanisms are used to conduct load testing and scalability.	Data integrity, confidentiality, and security are ensured.	Open Web Application Project (OWASP) and Zed Attack Proxy (ZAP) are used for vulnerability testing. Postman for API security testing.
Transaction Speed	Transaction speed was achieved using Hyperledger Fabric's enhanced consensus techniques (PBFT).	Average transaction delay is 150 milliseconds, with 100 transactions per second.	Apache JMeter was used to measure transaction speed.
Scalability	The system was designed to handle high simultaneous user traffic.	Supported up to 1,500 simultaneous users during peak load testing with stable system performance.	Load testing and scalability assessment were done using locust.
Storage Solution	Decentralized offline file storage using InterPlanetary File System (IPFS).	Allows for secure, distributed storing of large documents while maintaining metadata and hashes on the blockchain ledger.	IPFS Command Line Interface (CLI) tools are used for storage validation and manual data retrieval tests.
User Interface (UI)	Built using React.js for a single-page, dynamic application.	Provided user-friendly experience through real-time updates and seamless navigation.	UI feedback was measured using Usability Hub, while Google Lighthouse was used for performance audits.
Backend Architecture	The backend was powered by Node.js and integrated with Hyperledger Fabric through RESTful APIs.	Seamless interaction between the web application and blockchain network was ensured.	Postman and New Relic were used for API tests and backend performance monitoring.
Document Management	Aspects include upload, version control, retrieval, and integrity verification.	Real-time integrity checks utilizing blockchain hashes presented efficient document lifecycle management.	Manual testing and Postman are used to validate processes.
Consensus Mechanism	Practical Byzantine Fault Tolerance (PBFT) was adopted.	Transaction validation was dependable while preserving high security and performance.	Hyperledger Fabric CLI tools were used to monitor consensus performance.
System Reliability	Performance reliability was tested under varying workloads.	The average response time was 300 ms, ensuring steady performance even during large-scale activities.	Locust and Apache JMeter were used to test stress and reliability.
Cost Efficiency	Monthly operating costs are reduced due to decentralized storage and resource sharing.	Expenses were reduced to One Hundred Thousand Naira (₦100,000) monthly, with cheaper maintenance costs than traditional archive systems.	Cost analysis was measured using Amazon Web Services (AWS) pricing tools for hosted services and storage.
Auditability	Blockchain ledger stores immutable records of document transactions.	Provided a transparent, verifiable history of all document-related actions, ensuring compliance and auditability.	Hyperledger Fabric Explorer is used for transaction audits.
User Feedback	Feedback was collected from users during usability testing.	Feedback on simplicity, ease of navigation, and overall satisfaction with the interface was positive.	Google Forms was used to conduct surveys and interviews on user experience.

Interoperability	The system is designed to integrate with existing legacy systems through APIs.	Ensured easy migration of existing data and flawless interaction with existing systems.	Postman was used for API integration testing, and Swagger was used for API documentation.

4.5. Comparison of Traditional Systems and Document Management Systems Leveraging Blockchain

Table 2. Comparison of Traditional Systems and Document Management Systems Leveraging Blockchain

Feature	Traditional System	Blockchain-Based Document Management System
Data Integrity	The traditional system can be exposed to tampering, loss of records, and unauthorized access due to centralized storage.	The record is immutable and tamper-proof. A cryptographic hash is used to ensure data integrity.
Security	The system depends on traditional encryption and centralized access control, which is inadequate to ensure data protection.	Security of records is enhanced using decentralized storage with encryption and consensus techniques.
Access Control	Centralized storage is usually vulnerable to unauthorized access threats.	Due to the system being powered by Blockchain, it is built with decentralized permissions and role-based access managed through smart contracts.
Cost	The cost of maintenance, backups and recovery is usually high.	Lower costs can be achieved by using decentralized storage and reducing infrastructure needs.
Scalability	System performance can be negatively impacted due to increased data volume, limiting scalability.	The system supports high load and data distribution.
Transparency	Administrators usually manage audit trails, exposing the system to limited transparency.	The system is fully transparent and traceable, with immutable logs stored on the blockchain network.

4.6. System Performance Evaluation

The system performance is assessed with relevant evaluation tools and Performance Indicators, as shown in Table 2.

Table 3. Performance Evaluation

Performance Indicator	TOOL USED	VALIDATION
Network Performance	Hyperledger Caliper	The tool gives detailed measurements for transaction throughput and latency.
Chaincode Performance	Hyperledger Caliper	Assesses execution time and transaction processing speed.
Database Performance	MondoDB, Studio 3T	MongoDB monitors and optimizes database performance, while Studio 3T provides additional tools for querying, optimizing, and evaluating MongoDB.
Performance	JMeter	JMeter is an open-source load and performance measuring tool that tests web applications under various scenarios.
	New Relic	Provides real-time insights into system performance, assisting to discover and deal with performance bottlenecks.
	AppDynamics	This application performance monitoring solution provides detailed visibility into application performance, allowing for proactive monitoring and performance optimization.
Security	OWASP ZAP	Open-source security tool for detecting online application vulnerabilities, such as SQL injection and cross-site scripting.
	Burp Suite	Burp Suite is a comprehensive security testing tool for web applications, offering automated scanning and vulnerability management.
	OpenSSL	Provides robust encryption and security features for secure communication and data protection.
User Experience (UX)	Google Lighthouse	Audits performance, accessibility, and Search Engine Optimization (SEO) of web applications, providing actionable insights for improving UX.
	User Testing	Offers user feedback and usability testing to identify user pain points and improve satisfaction.
	Hotjar	Provides heat maps, session recordings, and user feedback tools to understand user behavior and experience.

Cost	IBM Cloud Cost Estimator	Helps estimate the cost of deploying and running applications on IBM Cloud, providing detailed cost breakdowns.
Fault Tolerance	Chaos Monkey	Test fault tolerance of web-based applications by intentionally causing failures to ensure system recovery and robustness. It helps ensure that the web-based application can withstand unexpected failures and remain operational.
	Resilience testing tools-Gremlin, Fault Injection Testing (FIT) frameworks	General tools for evaluating and improving system fault tolerance and resilience.
Benchmarking	Hyperledger Caliper	Providing comprehensive performance metrics and benchmarks.
Load Testing	JMeter	Suitable for load testing web applications and services, allowing simulation of various load conditions and performance analysis.
	Locust	Open-source load testing tool that simulates high user traffic, providing real-time performance metrics.

4.7. Major Security Concerns of the System

- *Critical Management Challenges:* Blockchain-powered systems depend majorly on cryptographic keys to secure transactions and data. The loss or compromise of the node's private key can lead to unauthorized access to stored records on the system.
- *Susceptibilities in Smart Contract:* If a smart contract is prone to unauthorized access to records, incorrect calculation of numerical values, and insufficient access control techniques, it can be exploited by attackers to tamper with document records or interrupt system processes.
- *Distributed Denial of Service (DDoS) attacks:* Access to the system's web interface could be disrupted by a Distributed Denial-of-Service (DDoS) attack, which could affect the functionality and availability of the system.

5. Conclusion

The system enhances security, efficiency, and reliability. Hyperledger Fabric's permissioned blockchain ensures only verified users access sensitive data, while its high throughput and low latency handle large volumes of documents and transactions efficiently.

The system's modular design allows for modifying and integrating components such as consensus procedures, identity management, and chain code, which automates document management tasks, including access control, versioning, and audit trails.

Private data collection and channels maintain confidentiality, while the tamper-proof ledger assures traceability and accountability. Furthermore, seamless integration with existing systems enables smooth data migration and interoperability, allowing enterprises to modernize their document management systems without affecting operations. Hyperledger Fabric's capabilities make it an excellent choice for a safe and efficient document archive management system.

5.1. Recommendations

Implement advanced encryption methods and multifactor authentication to protect against evolving threats and attacks.

Additionally, enhance the system's functionality and user experience by integrating features such as real-time collaboration tools, cloud storage solutions, and AI-driven document classification and retrieval.

5.2. System Limitations

The system is limited in managing large volumes of documents, as scalability concerns may emerge with growing users and documents.

As the network expands, substantial resources and optimization may be needed to sustain performance and responsiveness.

5.3. Suggestion for Further Studies

Further studies could explore the integration of the system with emerging technologies such as Artificial Intelligence (AI) and Machine Learning (ML). This could enhance capabilities like document classification, automated tagging, and intelligent search, providing more advanced and user-friendly features.

Further research could look into advanced encryption techniques, multifactor authentication, and intrusion detection systems to strengthen the system's entire security framework.

Funding Statement

The development and publication of this work were entirely funded by the authors, without any external financial support or sponsorship.

References

[1] Abdullah Ibrahim Shaban, Marwan Abdulhussein Farhan, and Saadaldeen Rashid Ahmed, "Building a Smart System for Preservation of Government Records in Digital Form," *International Congress on Human-Computer Interaction, Optimization and Robotic Applications*, Ankara, Turkey, pp. 1-6, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[2] Adam Nix et al., "Archival Research in the Digital Era," *Handbook of Historical Methods for Management*, Edward Elgar Publishing, pp. 156-17, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[3] Maria Pia Donato, "Introduction: Archives, Record Keeping and Imperial Governance, 1500-1800," *Journal of Early Modern History*, vol. 22, no. 5, pp. 311-326, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[4] Geoffrey Yeo, *Record-Making and Record-Keeping in Early Societies*, 1st ed., Routledge, pp. 60-89, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] João Reisa, and Nuno Melão, "Digital Transformation: A Meta-Review and Guidelines for Future Research," *Heliyon*, vol. 9, no. 4, pp. 1-19, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[6] Ali Sunyaev, "Web Services," *Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Technologies*, Springer, pp. 155-194, 2020. [Google Scholar] [Publisher Link]

[7] Ruofei Zhang, Mark Zhang, and Guangbiao Pu, "Web-based Medical Data Archive System," *Proceeding of 9th Annual Clinical Research Conference*, pp. 1-7, 2019. [Google Scholar]

[8] Saadia Malik et al., "Web-Based Archive Management and Student Guidance for Final Year Projects," *Journal of eLearning and Higher Education*, vol. 2018, pp. 1-11, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[9] Gaganjeet Singh Reen, Manasi Mohandas, and S. Venkatesan, "Decentralized Patient Centric e-Health Record Management System using Blockchain and IPFS," *IEEE Conference of Information and Communication Technology*, Allahabad, India, pp. 1-7, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[10] Mohammad Saidur Rahman et al., "A Novel Architecture for Tamper Proof Electronic Health Record Management System using Blockchain Wrapper," *ACM International Symposium on Blockchain and Secure Critical Infrastructure*, pp. 97-105, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[11] C. Komalavalli, Deepika Saxena, and Chetna Laroiya, "Overview of Blockchain Technology Concepts," *Handbook of Research on Blockchain Technology*, pp. 349-371, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] Ali Sunyaev, "Distributed Ledger Technology," *Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Technologies*, pp. 265-299, 2020. [Google Scholar]

[13] Jin Sun·et al., "A Blockchain-Based Framework for Electronic Medical Records Sharing with Fine-Grained Access Control," *Plos One*, vol. 15, no. 10, pp. 1-23, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Huaqun Guo, and Xingjie Yu, "A Survey on Blockchain Technology and its Security," *Blockchain: Research and Applications*, vol. 3, no. 2, pp. 1-15, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[15] Prasanth Varma Kakarlapudi, and Qusay H. Mahmoud, "Design and Development of a Blockchain-Based System for Private Data Management," *Electronics*, vol. 10, no. 24, pp. 1-22, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[16] Jayapriya Jayabalan, and N. Jeyanthi, "A Review on State-of-Art Blockchain Schemes for Electronic Health Records Management," *Cybernetics and Information Technologies*, vol. 24, no. 1, pp. 35-63, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[17] Gautami Tripathi, Mohd Abdul Ahad, and Gabriella Casalino, "A Comprehensive Review of Blockchain Technology: Underlying Principles and Historical Background with Future Challenges," *Decision Analytics Journal*, pp. 1-21, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[18] P.K. Paul et al., "Blockchain Technology and its Types—A Short Review," *International Journal of Applied Science and Engineering*, vol. 9, no. 2, pp. 189-200, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[19] Sumaira Johar et al., "Research and Applied Perspective to Blockchain Technology: A Comprehensive Survey," *Applied Sciences*, vol. 11, no. 14, pp. 1-42, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[20] Jing Xiao et al., "CE-PBFT: A High Availability Consensus Algorithm for Large-Scale Consortium Blockchain," *Journal of King Saud University-Computer and Information Sciences*, vol. 36, no. 2, pp. 1-13, 2024. [CrossRef] [Google Scholar] [Publisher Link]

[21] Karthik Kumar Vaigandla et al., "Review on blockchain Technology: Architecture, Characteristics, Benefits, Algorithms, Challenges and Applications," *Mesopotamian Journal of Cyber Security*, vol. 2023, pp. 73-84, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[22] Ziad Hussein, May A. Salama, and Sahar A. El-Rahman, "Evolution of Blockchain Consensus Algorithms: A Review on the latest Milestones of Blockchain Consensus Algorithms," *Cybersecurity*, vol. 6, pp. 1-22, 2023. [CrossRef] [Google Scholar] [Publisher Link]

[23] Sivleen Kaur et al., "A Research Survey on Applications of Consensus Protocols in Blockchain," *Security and Communication Networks*, vol. 2021, no. 1, pp. 1-22, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[24] Bahareh Lashkari, and Petr Musilek, "A Comprehensive Review of Blockchain Consensus Mechanisms," *IEEE Access*, vol. 9, pp. 43620-43652, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[25] Huanliang Xiong et al., "Research on Progress of Blockchain Consensus Algorithm: A Review on Recent Progress of Blockchain Consensus Algorithms," *Future Internet*, vol. 14, no. 2, pp. 1-24, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[26] Hui Pang et al., "Research on Practical Byzantine Fault Tolerant Algorithm Based on Trust Mechanism," *Journal of Computers*, vol. 33, no. 2, pp. 11-23, 2022. [CrossRef] [Google Scholar] [Publisher Link]

[27] Muhammad Nasir Mumtaz Bhutta et al., "A Survey on Blockchain Technology: Evolution, Architecture and Security," *IEEE Access*, vol. 9, pp. 61048-61073, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[28] Moyegbemi Ajinomisanghan, "*Implementation of a Web-Based e-Voting System Using Hyperledger Fabric*," Master's Thesis, Concordia University of Edmonton, pp. 1-38, 2023. [Google Scholar] [Publisher Link]

[29] Max Jonsson, and Eric Qvarnström, "A Performance Comparison on REST-APIs in Express.js, Flask and ASP.NET Core," *International Conference on Software Engineering and Data Science*, pp. 1-33, 2022. [Google Scholar] [Publisher Link]

[30] Célia Talma Gonçalves, Maria José Angélico Gonçalves, and Maria Inês Campante, "Developing Integrated Performance Dashboards Visualizations Using Power BI as a Platform," *Information*, vol. 14, no. 11, pp. 1-16, 2023. [CrossRef] [Google Scholar] [Publisher Link]