isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 6, Issue: 3,Special Issue: 4 ,Apr,2026 ,ISSN_NO: 2321-3337

Blockchain Enabled IoT Data Logger

Monika M[1],Divya TG[2],Anitha E[3],Umamageshwari A[4],Dr Kamal N [5]*

[1234]UG Scholar-Dept CSE, Grt Institute of Engineering and Technology Tiruttani, India.
[5]*Professor-Dept CSE, Grt Institute of Engineering and Technology Tiruttani, India. [1]
monikacse2022@grt.edu.in, divyacse2022@grt.edu.in
anithacse2022@grt.edu.in umamageshwari@grt.edu.in
[5*]**Corresponding Author: kamal.n@grt.edu.in**

Abstract

Data logging systems are widely used in industries and IOT applications to collect and store sensor data. However, traditional data logging systems store information in centralized databases which may lead to data tampering, security issues, and lack of transparency. To overcome these problems, this project proposes a Block chain Enabled IOT Data Logger. The system collects real-time sensor data using IOT devices connected to an ESP32 micro-controller.The collected data is securely transmitted and stored using Block-chain technology to ensure data integrity and immutability. Block-chain helps in preventing unauthorized modifications and provides a secure decentralized data storage mechanism. This system can be used in industrial monitoring, environmental monitoring, and smart city applications where secure and reliable data logging is required.

Keywords: *Blockchain, IoT, Data Logger, ESP32, Sensor Data, Security*

1.Introduction

The Internet of Things (IOT) has enabled smart devices to collect and transmit data continuously from different environments. Data loggers are commonly used to record sensor information such as temperature, humidity, and other environmental parameters. These systems are widely used in industries, agriculture, healthcare, and environmental monitoring.However, traditional data logging systems store data in centralized servers which may lead to security vulnerabilities. Unauthorized users may modify or delete important data,

leading to loss of reliability and trust.Block-chain technology provides decentralized and secure way of storing information. Once data is stored in block-chain, it cannot be altered or deleted. By integrating IoT devices with block-chain, the collected sensor data can be securely stored and verified.This project proposes a Block-chain Enabled IOT Data Logger that collects real-time data using IOT sensors and stores it securely in block-chain to ensure transparency, security, and data integrity.

2.Related Work

Recent research has explored the integration of block-chain technology with IOT systems to enhance data security, integrity, and decentralization. Traditional IOT data logging frameworks rely on centralized cloud platforms, which introduce vulnerabilities such as single-point failures, data tampering, and limited transparency among stakeholders. Researchers have proposed decentralized architectures to overcome these challenges by leveraging distributed ledgers and cryptographic mechanisms.

Several studies have investigated block chain-based data storage for IOT environments. Alcaraz et al. highlighted the potential of block chain to provide tamper-resistant logging in industrial IOT (IIoT) settings, demonstrating improved traceability and auditability compared to centralized solutions . Similarly, Dorri et al. proposed a lightweight block-chain framework tailored for IOT devices, wherein smart contracts manage access control policies, enabling secure and transparent data exchanges between devices However, these frameworks often encounter scalability constraints due to high computational costs and limited resource capacities of IOT hardware.

To address scalability and latency, approaches combining edge computing and block-chain have been explored. Nguyen et al. proposed a hybrid architecture where data pre-processing at edge nodes reduces network overhead before committing records to the block-chain, increasing system efficiency and lowering transaction latency. Edge-assisted block-chain logging has also been investigated in agricultural monitoring systems, where sensor data is aggregated and filtered at fog nodes prior to block-chain storage, thus optimizing bandwidth usage. Despite improvements in performance, these works typically emphasize data collection efficiency and less on robust encryption or timestamp integrity.

Smart contract-enabled authentication mechanisms are another focus in the literature. Christidis and Devetsikiotis demonstrated how Ethereum smart contracts could enforce secure IOT data logging by verifying device identities and automating validation processes. While this improves trustworthiness, the overhead of executing smart contracts remains a concern for real-time applications with high data throughput.

In the healthcare domain, Zhang et al. developed a block chain-based medical data platform to ensure immutability and patient privacy, integrating attribute-based encryption to satisfy confidentiality requirements

Although effective for sensitive data, the approach may not be directly transferable to resource-constrained IOT environments without modifications to reduce encryption complexity.

Compared to existing work, the proposed system distinguishes itself by combining edge-level pre-processing, robust encryption, and smart contract-driven logging to securely and efficiently record IOT data on the block-chain. The use of lightweight edge devices such as Raspberry Pi and ESP32 enables both pre-processing and encryption before data submission, allowing for reduced latency and lower computational overhead. Furthermore, the integration of smart contracts ensures authenticated and immutable data storage, improving trust among decentralized stakeholders without sacrificing performance.

3.Objective

The primary objective of this research is to develop a block chain-enabled **IOT data logging system** that ensures secure, reliable, and transparent storage of sensor-generated data. The system aims to collect real-time data from IOT sensors such as temperature, humidity, pressure, and device status, and securely record this information using block-chain technology. By integrating IOT devices with a decentralized block-chain ledger, the framework ensures that once the data is recorded, it becomes immutable and resistant to tampering or unauthorized modifications

4.Proposed System

The proposed system implement Block-chain enabled IOT Data Logger that continuously monitors critical parameters such as temperature, humidity, pressure, and machine status indicators including normal operation, overheating, and excessive vibration. IoT sensors deployed in the system collect real-time data from the physical environment.

Each sensor reading is systematically captured and formatted into structured datasets to ensure consistency and reliability in data processing.

The collected datasets are then encrypted using Cryptographic techniques to ensure confidentiality and data integrity during transmission. After encryption, the data is securely recorded on a block-chain network. The block-chain ensures that every data entry is time-stamped, traceable, and tamper-resistant due to its decentralized and immutable ledger structure. Each block contains a cryptographic hash of the previous block, thereby preventing unauthorized modification of stored records.

The system back-end maintains a secure distributed ledger of all IOT logs, enabling authorized users to access, verify, and audit the recorded data without compromising its integrity. Smart contracts are employed to authenticate devices and validate data transactions before they are permanently added to the block-chain. By integrating IOT sensing, encryption mechanisms, and block-chain technology, the proposed system provides a secure, transparent, and reliable solution for real-time data logging in applications such as industrial monitoring

5.Architecture Diagram

This architecture explains how IOT sensor data is collected, processed, secured, and stored using block-chain technology. In the IOT Data Acquisition Layer, sensors such as temperature, humidity, pressure, and vibration sensors collect real-time environmental data. This data is then sent to the Edge Processing and Encryption Layer, where an ESP32 micro controller processes the sensor data, adds a timestamp, and secures it using AES encryption. After that, the data moves to the IOT Gateway Server Layer, where the system validates the device and converts the encrypted data into a secure hash using the SHA-256 algorithm. The hashed data is then stored in the Block-chain Layer, where blocks are created and the data is recorded in an immutable ledger, making it tamper-proof and secure. Finally, in the Application Layer, the stored data is displayed to users through dashboards, graphs, and monitoring tools, allowing users to analyze and track the IoT data

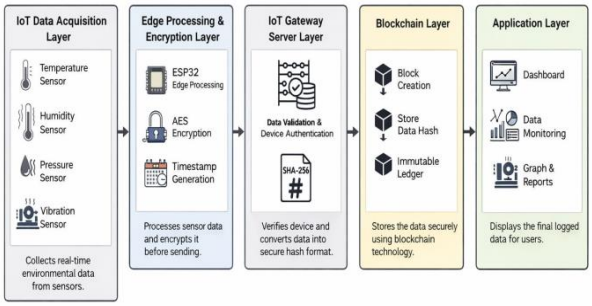

Fig.5:ArchitectureDiagram

6.Algorithm

The proposed system implements a **hybrid cryptographic approach combining AES-256 encryption and SHA-256 hashing** to ensure secure transmission and storage of IoT sensor data. This algorithm guarantees confidentiality, integrity, and tamper-resistance before data is recorded in the blockchain ledger.

Initially, sensor data collected from IoT devices such as temperature, humidity, pressure, and vibration sensors is formatted and timestamped at the edge device. The processed data is then encrypted using the **Advanced Encryption Standard (AES-256)** symmetric encryption algorithm. AES-256 converts the plain sensor data into ciphertext using a secret encryption key, preventing unauthorized entities from accessing the original information during transmission.

After encryption, the system generates a **cryptographic hash using the SHA-256 algorithm**. The hash function produces a fixed-length unique digital fingerprint for each encrypted dataset. This hash value ensures data integrity, as any modification in the original data will produce a completely different hash value.

The encrypted data along with the generated hash is transmitted to the **IoT gateway server**, which validates the device identity and forwards the transaction to the blockchain network. The blockchain layer stores the generated hash as part of a block transaction, ensuring that the data becomes **immutable and tamper-proof** once recorded.

This cryptographic mechanism enhances security by protecting sensitive IoT data from interception, unauthorized access, and manipulation while maintaining transparency through blockchain-based storage.

7.Implementation

The proposed system implements a **secure IoT data logging framework integrated with block-chain technology** to ensure reliable and tamper-proof storage of sensor data. The architecture is divided into multiple layers that handle data acquisition, pre-processing, secure transmission, decentralized storage, and real-time monitoring. Each layer performs a specific function to maintain data integrity, confidentiality, and transparency across the entire system.

7.1 IoT Data Acquisition Layer

The **IoT Data Acquisition Layer** is responsible for collecting real-time environmental and machine-related data from various sensors. In the proposed system, sensors such as **temperature, humidity, pressure, and vibration sensors** are deployed to continuously monitor environmental conditions and system status. These sensors capture physical parameters and convert them into digital signals that can be processed by the microcontroller.

The collected sensor data is transmitted to the processing unit at regular intervals to ensure continuous monitoring. This layer forms the foundational component of the system, as it provides accurate real-time data required for further processing, security validation, and storage within the blockchain network.

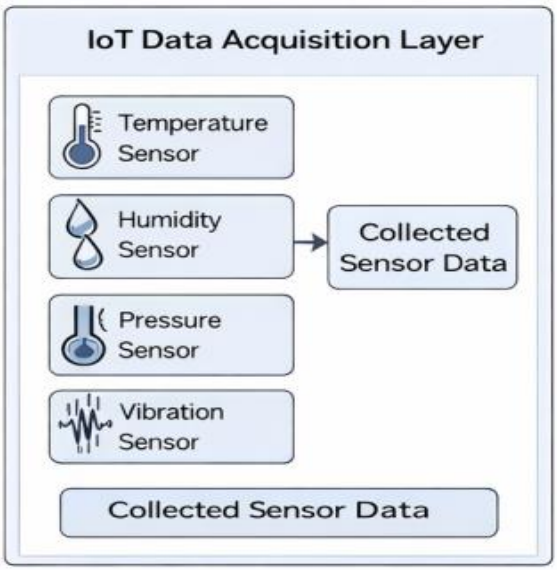

Fig 7.1 IoT Data Acquisition Layer

7.2 Edge Processing and Encryption Layer

The **Edge Processing and Encryption Layer** performs preliminary data processing and security operations before transmitting the sensor data to the block-chain network. In this layer, a microcontroller such as **ESP32** performs edge-level data processing to reduce latency and enhance efficiency.

The raw sensor readings undergo several operations including data validation, formatting, and timestamp generation to ensure consistent data structure. Additionally, the system applies AES encryption to protect the sensor data from unauthorized access during transmission.

Volume: 6, Issue: 3,Special Issue: 4 ,Apr,2026 ,ISSN_NO: 2321-3337

This layer also prepares the data for blockchain integration by generating structured datasets.By performing these operations at the edge, the system improves response time and reduces computational load on centralized components

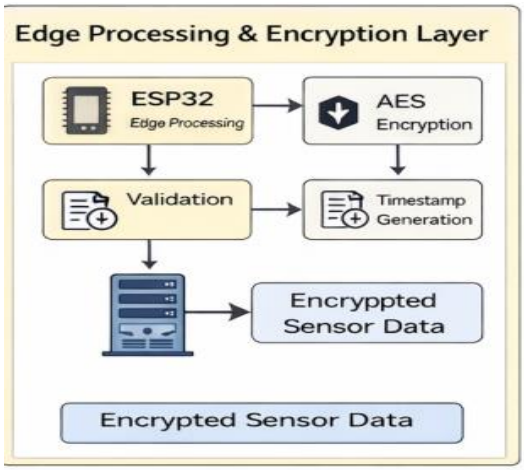

Fig 7.2 Edge Processing and Encryption Layer

7.3 IoT Gateway Server Layer

The **IoT Gateway Server Layer** acts as the intermediary between the edge device and the blockchain network. This layer is responsible for receiving the processed sensor data and verifying its authenticity before forwarding it to the blockchain.

The gateway performs **device validation and authentication** to ensure that only authorized IoT devices can submit data. Additionally, it generates a **cryptographic hash using SHA-256**, which creates a unique fingerprint for each dataset. This hash ensures that any modification to the data can be easily detected.The gateway then converts the data into a blockchain-compatible transaction format and sends it to the blockchain network for secure storage.

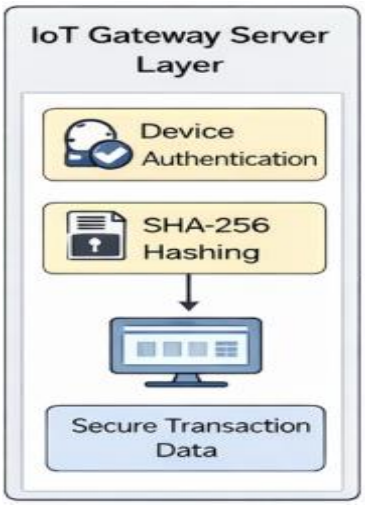

Fig 7.3 IoT Gateway Server Layer

7.4 Blockchain Layer

The **Blockchain Layer** provides decentralized and tamper-proof storage for IoT sensor data. In the proposed system, blockchain technology is used to ensure that once sensor data is recorded, it cannot be altered or deleted.Each dataset received from the gateway is converted into a **blockchain transaction**, which is validated and added to the blockchain ledger. The blockchain stores the **hash of the data**, ensuring data integrity while maintaining an immutable record of all sensor logs.This decentralized storage mechanism eliminates single points of failure and ensures transparency, security, and trust in the data logging process.

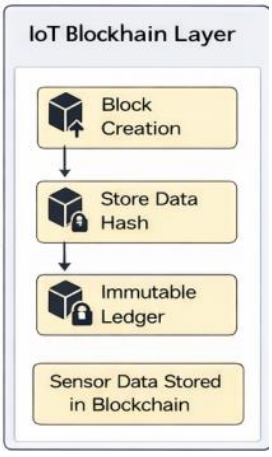

Fig 7.4.IoT Blockchain Layer

isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 6, Issue: 3,Special Issue: 4 ,Apr,2026 ,ISSN_NO: 2321-3337

7.5 Application Layer

The **Application Layer** provides the user interface for monitoring and analyzing the sensor data stored in the blockchain. This layer retrieves verified records from the blockchain network and displays them through a **dashboard interface**.The dashboard allows users to view **real-time data monitoring, graphical visualizations, and system reports**. These visual insights help users track environmental conditions and analyze system performance efficiently.This layer enhances usability by providing a clear and interactive representation of the collected data, enabling informed decision-making based on trusted sensor records.

Fig 7.5 Application Layer

8.Experimentatal Results

The experimental evaluation of the proposed **Blockchain Enabled IoT Data Logger** was conducted to verify the functionality of real-time sensor data acquisition, secure transmission, blockchain storage, and application-level visualization. The system was implemented using IoT sensors connected to an ESP32 microcontroller, with blockchain integration for secure and tamper-proof data storage.

Physical Device

The setup consists of multiple sensors connected to a microcontroller through jumper wires for collecting environmental data

.At the center of the setup is the **ESP32 microcontroller board**, which acts as the main processing unit. The ESP32 is responsible for reading sensor values, preprocessing the data, and transmitting it to the system for secure storage and monitoring.

Two sensor modules are connected to the microcontroller. One module is used for **environmental sensing such as temperature and humidity**, while the other module is used for **detecting environmental conditions such as vibration or sound levels**. These sensors capture real-time environmental data and send the readings to the ESP32 through the connected jumper wires.

The sensors and microcontroller are interconnected using **color-coded jumper wires**, which provide power supply, ground connection, and data communication channels between the components. The ESP32 processes the collected

sensor readings and prepares them for transmission to the application layer where the data is encrypted and stored securely using blockchain technology.This hardware configuration forms the **IoT Data Acquisition Layer** of the proposed system, enabling continuous monitoring of environmental conditions and providing reliable input data for secure blockchain-based logging.

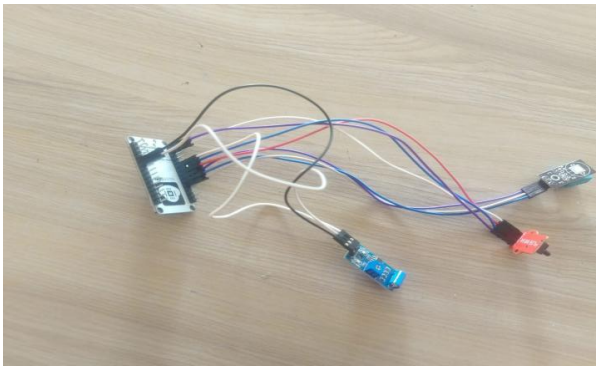

Fig 8 IoT sensor modules connected to the ESP32 microcontroller for real-time environmental data acquisition.

isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 6, Issue: 3,Special Issue: 4 ,Apr,2026 ,ISSN_NO: 2321-3337

8.1 Sensor Data Acquisition

The first experiment verifies the real-time acquisition of environmental data from IOT sensors. Sensors connected to the ESP32 continuously measure parameters such as temperature, humidity, and device status. The sensor readings are transmitted through serial communication and processed by the system backend.

Fig 8.1 Sensor data Acquisition

8.2 User Registration

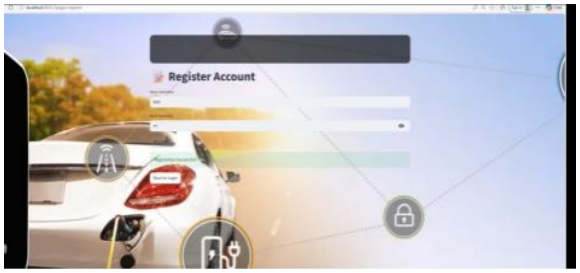

Fig 8.2 User Registration

The system provides a user registration that allows new users to create an account in order to access the IoT monitoring system. During registration, users provide details such as email, and password. These details are securely stored in the system database.This ensures that only authenticated users can access the system and view the recorded sensor data.

8.3 Login

The interface contains fields for **username and password**, where registered users can enter their credentials.

After entering valid login details, users can click the **Login** button to access the system. If a user does not already have an account, they can create one by selecting the **"Go to Register"** option provided on the page.

This authentication mechanism ensures that only **authorized users** can access the IoT monitoring system and retrieve sensor data stored in the blockchain ledger. The login page acts as the first security layer in the system, preventing unauthorized access and protecting sensitive IoT data

Fig 8.3 Login interface of the Smart Machinery monitoring system used for secure user authentication

8.4 Dashboard Monitoring

After authentication, the user is redirected to the **dashboard page**, where various environmental and machine parameters are displayed. The system continuously collects sensor readings such as **temperature, humidity, pressure, and vibration levels** from the IoT sensors connected to the ESP32 microcontroller.The dashboard presents these parameters in both **numerical and graphical formats** using gauge-style visualization.

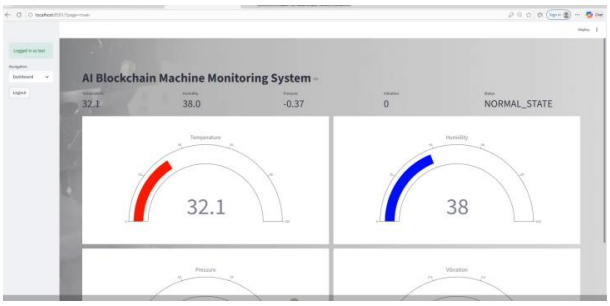

Fig 8.4 Real-time monitoring dashboard displaying IoT sensor

These values are updated in real time, allowing users to monitor machine conditions and environmental parameters efficiently. The graphical gauges help users quickly interpret sensor readings and detect abnormal conditions.

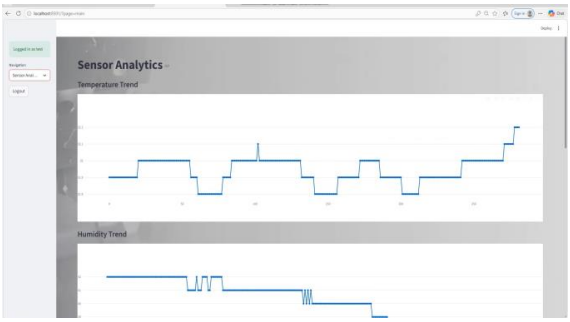

Fig 8.4.1 Sensor data Analytics

The dashboard displays **historical trends of environmental parameters** collected from sensors connected to the ESP32 microcontroller. The collected sensor data is continuously recorded and securely stored using blockchain technology to ensure data integrity and transparency.

The interface presents the sensor readings in the form of **line graphs**, allowing users to analyze variations in environmental conditions over time.

Temperature Trend

The first graph represents the **temperature trend**, showing the variation in temperature values over a specific time interval.

The plotted line graph helps users observe changes in temperature and identify abnormal fluctuations that may affect machine performance or environmental stability.

Fig. 8.4.2: Temperature trend graphs showing variations in machine conditions over time.

Humidity Trend

The second graph illustrates the **humidity trend**, displaying how humidity levels change over time. This visualization enables users to monitor environmental conditions and evaluate system behavior under different humidity levels.

Fig. 8.4.3: Humidity trend graphs showing variations in machine conditions over time.

The graphical representation of sensor data allows users to easily track historical patterns, analyze system performance, and detect unusual conditions. This analytics feature improves monitoring efficiency and supports informed decision-making in industrial and IoT environments.

Pressure Trend

The upper graph represents the **pressure trend**, showing variations in pressure values recorded by the pressure sensor over time. The plotted data points indicate how pressure levels fluctuate during system operation. Occasional spikes in the graph represent sudden changes in pressure conditions, which may indicate environmental variations or machine operational changes.

isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 6, Issue: 3,Special Issue: 4 ,Apr,2026 ,ISSN_NO: 2321-3337

Vibration Trend

The lower graph displays the **vibration trend**, which is used to monitor the vibration levels of the monitored machine. Under normal operating conditions, the vibration values remain close to zero, indicating stable machine performance. However, occasional spikes in the graph indicate temporary increases in vibration levels, which may occur due to machine movement, operational load changes, or external disturbances.

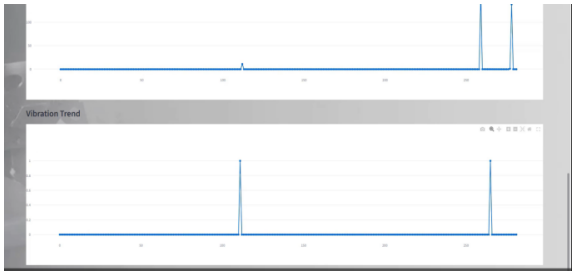

Fig. 8.4.4: Pressure and vibration trend graphs showing variations in machine conditions over time.

The graphical analysis helps users identify abnormal behavior in machine performance by observing sudden spikes or deviations in sensor values. By continuously monitoring these parameters, the system enables early detection of potential faults and supports predictive maintenance.

All sensor readings displayed in the analytics dashboard are collected through IoT sensors, processed by the ESP32 microcontroller, and securely stored in the blockchain network to ensure data integrity and transparency.

8.5 Blockchain Ledger Viewer Interface

The **Blockchain Ledger Viewer interface** of the Blockchain Monitoring System allows users to view the blockchain records where IoT sensor data is securely stored. The interface displays the blockchain ledger in a tabular format, showing each block and its associated transaction details.

Each row in the table represents a **block in the blockchain**, containing information such as:

- **Hash value** – a unique cryptographic identifier generated for each block
- **Index** – the position of the block within the blockchain
- **Previous hash** – the hash of the previous block that links blocks together
- **Timestamp** – the exact time when the data was recorded
- **Sensor data values** including humidity, pressure, temperature, vibration, and system status

The **previous hash field** connects each block with the preceding block, forming a continuous chain. This structure ensures that if any block is modified, the hash values will change, making tampering easily detectable. This mechanism guarantees **data integrity and immutability** within the blockchain network.

Below the ledger table, the interface displays the **latest hash value**, which represents the most recent block added to the blockchain. This value is useful for verifying the integrity of the latest transaction recorded in the system.

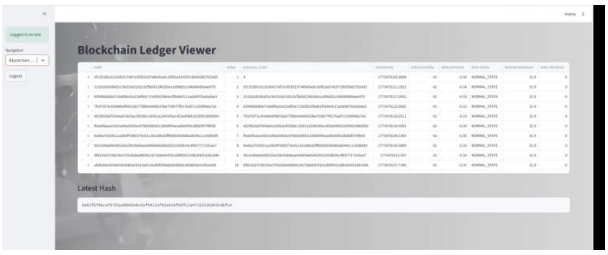

Fig.8.5 Blockchain ledger viewer displaying stored IoT sensor data and block hash information.

Through this module, users can transparently verify all stored IoT sensor records and ensure that the logged data has not been altered. The blockchain ledger viewer therefore provides **secure, traceable, and tamper-proof monitoring of machine and environmental data**.

isrjournals

International Journal of Advanced Research in Computer Science Engineering and Information Technology

Volume: 6, Issue: 3,Special Issue: 4 ,Apr,2026 ,ISSN_NO: 2321-3337

8.6 Machine Fault Prediction Interface

The image shows the **Machine Fault Prediction module** of the AI Blockchain Machine Monitoring System. This interface is used to analyze the collected IoT sensor data and predict the operational condition of the monitored machine.

After logging into the system, users can navigate to the **Fault Prediction section** from the dashboard. This module processes sensor readings such as temperature, humidity, pressure, and vibration levels to determine the current health status of the machine.

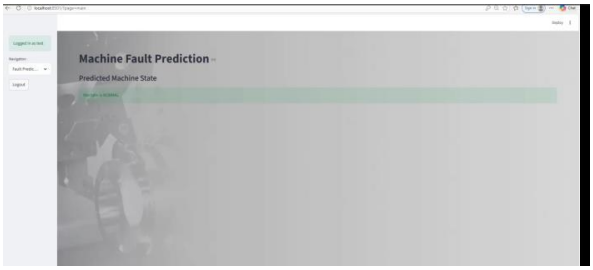

Fig 8.6 Machine fault prediction interface displaying the predicted operational status of the monitored system.

The interface displays the **Predicted Machine State**, which indicates whether the machine is operating under normal conditions or experiencing abnormal behavior. In the image, the system output shows **"Machine is NORMAL"**, meaning that all sensor values are within safe operational thresholds and no faults are detected.

The prediction result is presented in a highlighted notification panel, making it easy for users to quickly understand the machine's condition. This feature helps in **early detection of potential faults**, enabling timely maintenance and preventing unexpected machine failures.

By combining IoT sensor monitoring with intelligent analysis and blockchain-based secure data logging, the system provides a reliable platform for **predictive maintenance and real-time machine monitoring**.

8.7 System Activity Logs Interface

The activity logs are presented in a **tabular format**, where each row represents a recorded event in the system. The table includes several important fields such as:

- **Index** – the sequential position of the record within the log
- **Timestamp** – the exact time at which the transaction or system event occurred
- **Hash value** – a unique cryptographic identifier associated with the stored data block

These logs help track all system activities and blockchain transactions, ensuring transparency and traceability of the recorded sensor data. The **hash values** displayed in the log represent the cryptographic output generated when the sensor data is processed and added to the blockchain ledger.

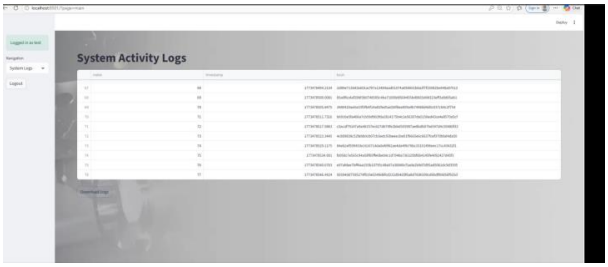

Fig 8.7 System activity logs interface displaying blockchain transaction records and timestamps.

The interface also includes an option to **download the activity logs**, allowing users to export system records for further analysis, auditing, or backup purposes.This module plays an important role in maintaining system accountability by providing a chronological record of blockchain operations and sensor data transactions. It ensures that all system activities can be monitored, verified, and audited whenever required.

8.8 Secured Data Output for Analysis

The image shows the **final secured data output generated by the Blockchain Enabled IoT Data Logger system**, displayed in a spreadsheet format for further analysis. The recorded data represents blockchain transaction records created after processing the sensor data collected from IoT devices.

Each row in the spreadsheet represents a **block entry generated by the system**, containing key information such as:

Index – the sequential position of the block in the blockchain ledger

Timestamp – the exact time when the sensor data was recorded

Hash value – a unique cryptographic identifier generated for the block

The **hash values** shown in the table are produced using cryptographic hashing algorithms, ensuring that each data record is uniquely identifiable and securely linked to the blockchain ledger. These hashes help maintain **data integrity and immutability**, as any modification to the stored data would result in a completely different hash value.

The spreadsheet format allows the system administrators or researchers to **export, store, and analyze the secured blockchain data** for further monitoring, auditing, or performance evaluation. By analyzing these records, users can verify the authenticity of the stored sensor data and ensure that the logging process remains transparent and tamper-proof.

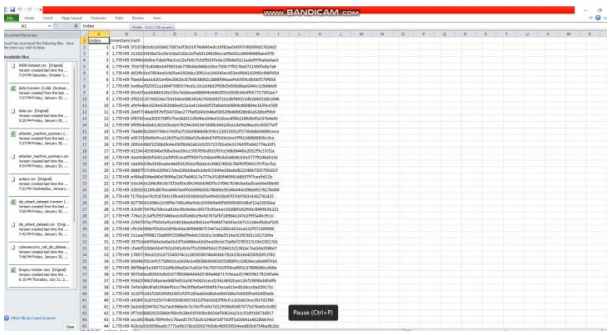

Fig. 8.8 Final secured blockchain data output exported for further analysis and verification.

This output confirms that the proposed system successfully records IoT sensor data in a **secure, traceable, and verifiable blockchain-based structure**, enabling reliable data analysis and system monitoring.

9.Conclusion and Future Work

This work presented a **Blockchain Enabled IoT Data Logger system** designed to securely collect, process, and store real-time sensor data for machine monitoring applications. The system integrates IoT sensors, an ESP32 microcontroller, cryptographic security mechanisms, and blockchain technology to ensure reliable and tamper-proof data logging. Environmental and machine parameters such as temperature, humidity, pressure, and vibration are continuously monitored and securely transmitted using **AES encryption and SHA-256 hashing**. The processed data is stored in a decentralized blockchain ledger, ensuring data integrity, transparency, and immutability. A web-based dashboard allows authorized users to monitor sensor data, analyze trends, verify blockchain records, and view system activity logs. Experimental results demonstrate that the system successfully provides secure real-time monitoring and protects sensor data from unauthorized modifications.

In future work, the system can be enhanced by integrating **advanced machine learning algorithms** for more accurate fault prediction and predictive maintenance. Scalability can be improved by adopting **advanced blockchain architectures** to handle large-scale IoT deployments. Additionally, integrating **cloud services, mobile monitoring applications, and additional IoT sensors** can further enhance system functionality, accessibility, and performance for real-world industrial environments.

Volume: 6, Issue: 3,Special Issue: 4 ,Apr,2026 ,ISSN_NO: 2321-3337

Reference

[1] Dorri, A., Kanhere, S. S., and Jurdak, R., "Blockchain in Internet of Things: Challenges and Solutions," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8076–8094, Oct. 2019.

[2] Christidis, K., and Devetsikiotis, M., "Blockchains and Smart Contracts for the Internet of Things," *IEEE Access*, vol. 4, pp. 2292–2303, May 2016.

[3] Reyna, A., Martín, C., Chen, J., Soler, E., and Díaz, M., "On Blockchain and Its Integration with IoT: Challenges and Opportunities," *Future Generation Computer Systems*, vol. 88, pp. 173–190, Nov. 2018.

[4] Dorri, A., Steger, M., Kanhere, S. S., and Jurdak, R., "Blockchain: A Distributed Solution to Automotive Security and Privacy," *IEEE Communications Magazine*, vol. 55, no. 12, pp. 119–125, Dec. 2017.

[5] Zheng, Z., Xie, S., Dai, H., Chen, X., and Wang, H., "An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends," *IEEE International Congress on Big Data*, pp. 557–564, June 2017.

[6] Zhang, Y., Kasahara, S., Shen, Y., Jiang, X., and Wan, J., "Smart Contract-Based Access Control for the Internet of Things," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 1594–1605, April 2019.

[7] Khan, M. A., and Salah, K., "IoT Security: Review, Blockchain Solutions, and Open Challenges," *Future Generation Computer Systems*, vol. 82, pp. 395–411, May 2018.

[8] Wood, G., "Ethereum: A Secure Decentralised Generalised Transaction Ledger," *Ethereum Project Yellow Paper*, 2014.

[9] Nakamoto, S., "Bitcoin: A Peer-to-Peer Electronic Cash System," 2008. Available: https://bitcoin.org/bitcoin.pdf

[10] Li, X., Jiang, P., Chen, T., Luo, X., and Wen, Q., "A Survey on the Security of Blockchain Systems," *Future Generation Computer Systems*, vol. 107, pp. 841–853, June 2020.

[11] Roman, R., Najera, P., and Lopez, J., "Securing the Internet of Things," *Computer*, vol. 44, no. 9, pp. 51–58, Sept. 2011.

[12] Atzori, L., Iera, A., and Morabito, G., "The Internet of Things: A Survey," *Computer Networks*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[13] Stallings, W., *Cryptography and Network Security: Principles and Practice*, 7th ed., Pearson Education, 2017.

[14] Kumar, N., and Mallick, P. K., "Blockchain Technology for Security Issues and Challenges in IoT," *Procedia Computer Science*, vol. 132, pp. 1815–1823, 2018.

[15] Zheng, Z., Xie, S., Dai, H., Chen, X., and Wang, H., "Blockchain Challenges and Opportunities: A Survey," *International Journal of Web and Grid Services*, vol. 14, no. 4, pp. 352–375, 2018.

[16] Xu, L. D., He, W., and Li, S., "Internet of Things in Industries: A Survey," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 4, pp. 2233–2243, Nov. 2014.

[17] Casino, F., Dasaklis, T. K., and Patsakis, C., "A Systematic Literature Review of Blockchain-Based Applications: Current Status, Classification and Open Issues," *Telematics and Informatics*, vol. 36, pp. 55–81, Mar. 2019

Volume: 6, Issue: 3,Special Issue: 4 ,Apr,2026 ,ISSN_NO: 2321-3337