STHChain: A Scalable IoT-Fog-Blockchain Architecture with Lightweight Consensus for Secure Tourist Healthcare

***Abstract*-- Tourists' healthcare maintenance, especially during the outbreak of infectious diseases, is a major challenge in the tourism industry. Ensuring accurate tourists' healthcare data requires data collected using Internet of Things (IoT) devices within a smart city. Additionally, data integrity is preserved through storage on a blockchain platform to maintain reliability. Some works have addressed blockchain-based healthcare solutions for smart tourists, nevertheless many aspects require further investigation. Therefore, we propose a lightweight and scalable blockchain-based smart tourists healthcare model, called STHChain to present welfare services. The model first configures a hierarchical network of IoT devices, edge, and fog computing layers to collect, preprocess, and store tourists' healthcare data. Subsequently, a smart contract defines formal Role-Based Access Control mechanisms for the model's components, including sensor devices, edge nodes, fog nodes, hotels/hostels, welfare centers, and tourists. This contract also executes a deterministic function to classify tourists' health status based on sensor data, which governs their access to welfare service. Finally, healthcare data are included in blocks and verified using a proposed Raft-based Clustering consensus protocol called RaC to add the chain of blocks. The energy consumption and throughput of the model is evaluated using iFogSim simulator, demonstrating that it is energy-efficient, scalable and suitable for real-time tourist healthcare monitoring.**

Keywords: **Blockchain, Healthcare, Internet of Things, IoT, Smart City, Smart Tourist, Tourism, Welfare Services.**

1. INTRODUCTION

The outbreak of infectious diseases in recent decades has caused significant socioeconomic damage to the tourism industry, challenging its management [1]. To address this challenge, the concept of the smart tourist based on the smart city has been introduced [2]. Tourists' healthcare maintenance is a critical concern in the tourism industry, where services provided to tourists must be tailored to their healthcare status. To present appropriate services, environmental conditions must also be considered [3]. A comprehensive investigation of the tourism industry and environmental conditions plays a pivotal impact in enhancing tourists' healthcare. In recent years, smart cities have evolved to improve service quality. A smart city establishes a new urban paradigm offering intelligent living-environment services, intelligent decision-making, and optimal resource management, such that these services address challenges related to treatment costs [4], resource availability [5], and remote monitoring [6]-[8]. In a smart city, the latest digital technologies are employed to enhance healthcare outcomes [9]. Since the onset of the Covid-19 pandemic in 2019, smart-city technologies have been utilized to provide precise, error-free data [10] for treating pandemics [11] and preventing their spread [12]. Accurate data enable timely disease detection, thereby reducing treatment costs [4]. Conversely, smart services face security challenges for securely storing data, including authentication, unauthorized access, energy consumption, and data integrity [13], [14]. In recent years, blockchain technology has been employed to preserve healthcare data integrity [15], [16]. Therefore, we present a lightweight and scalable blockchain-based smart tourists healthcare model, called STHChain, including the following components:

- Hotels and hostels are equipped with Internet of Things (IoT) devices in smart neighborhoods, and tourists' healthcare data are collected by these devices.
- A hierarchical network is established comprising sensor, edge, and fog nodes located at health centers and hotels/hostels to store tourists' healthcare data.
- A smart contract uses tourists' healthcare data to determine their healthcare status for access to welfare services in the smart city. Access control for tourist services is then issued by the smart contract.
- Healthcare data of tourists are stored in blocks. The proposed consensus protocol uses clustering of fog nodes located at health centers and employs Raft consensus mechanism to verify blocks and append them to the chain of blocks.

The primary contributions of this work are outlined below:

- We propose an integrated IoT-Edge-Fog and Blockchain architecture tailored for smart tourism, reducing latency and energy overhead.
- A smart contract implements a formal RBAC model and a deterministic health-status function for automated, auditable access control.
- A two-tier pseudonymization scheme decouples tourists' real identities from on-chain health records, ensuring privacy-by-design.
- The novel Raft-based Clustering (RaC) consensus protocol improves scalability and energy efficiency through localized fog-node verification clusters.

The remainder of the paper is comprised: Section 2 surveys IoT-based healthcare research and the challenges facing the tourism industry. Section 3 proposes the blockchain-based healthcare model for smart tourists. Section 4 provides analysis and evaluation of the proposed model. Section 5 concludes with results and outlines future work.

2. Literature Review

In this section, the previous work related to the challenges facing the tourism industry and IoT-based healthcare research is reviewed.

2-1. Tourism Industry Challenges

The tourism industry faces challenges related to the economy, international politics, cultural differences among countries, and tourists' healthcare, as illustrated in Figure 1.

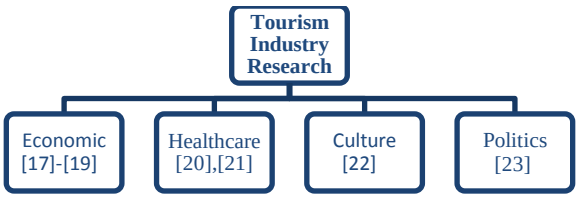

Fig. 1. Previous research in the area of tourism industry challenges.

Figure 1 shows several works in relation to challenges facing the tourism industry. Some researchers propose a green economy system aimed at addressing high energy consumption, water consumption, and habitat destruction in tourist destinations [17]. The study also examines the promotion of tourism culture within the social policies of the studied community [17]. Other researchers employ a digital economy framework, based on digital goods on the Internet and web-based applications, to investigate the economic challenges in the tourism industry [18]. The individual and collective dimensions of tourism that influence development and economic fluctuations are discussed in another study [19]. In recent years, the domain of tourists' healthcare has emerged to provide medical services to patients [20]. A substantial portion of people from developing countries travels to developed countries to continue their treatment. Accordingly, several efforts have been undertaken, for example, in Nigeria, to preserve the tourists' healthcare [20]. In India, healthcare facilities for tourists such as highway hospitals, mobile pharmacies, and applications for recording medical services have expanded, contributing to the mitigation of tourists' healthcare challenges in India [21]. Consequently, tourists' healthcare contributes to the prosperity of not only the tourism industry but also the broader regional economy. A recent systematic review of smart tourism technologies [34] highlights the growing role of IoT and data-driven services in enhancing tourist satisfaction and operational efficiency, underscoring the need for integrated technical frameworks that can deliver real-time, health-aware services. In another prominent research [22], religious beliefs in certain countries (e.g., Iran) that attract tourists were investigated. The researchers have shown that cultural and religious aspects of Iran, and the rich cultural heritage of other countries, influence the tourism industry of those societies. In another work [23], researchers studied current tourism policies and challenges on the island of Cyprus and concluded that policy changes in tourism are essential.

2-2. IoT-based Healthcare Systems

As discussed in the previous section, maintaining tourists' healthcare is a crucial factor in boosting the prosperity of the tourism industry. Accordingly, this subsection reviews the approaches proposed in IoT-based healthcare, as depicted in Figure 2.

Figure 2 illustrates some studies conducted in the area of IoT-based healthcare focused on secure storage and sharing of data using edge and fog computing [24]-[26]. In these studies, providing storage space through edge and fog computing not only reduces network traffic but also addresses many challenges inherent in cloud computing. Secure storage and sharing of healthcare data are of critical importance. if these data are compromised, the reliability of the healthcare system is undermined, with serious consequences for patients' healthcare.

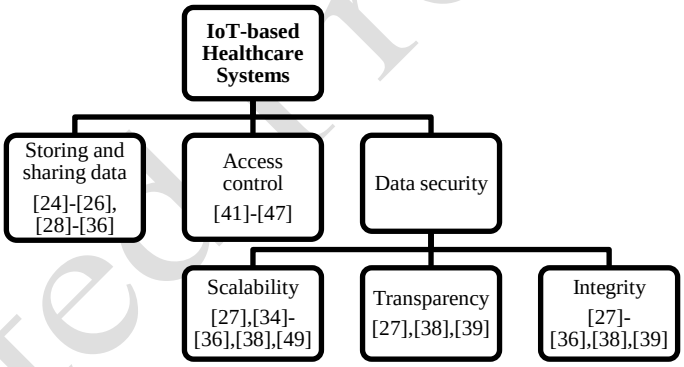

Fig. 2. Prominent research in the area of IoT-based healthcare.

Client–server and cloud-based healthcare data management systems face issues such as single point of failure, data privacy, centralized data monitoring, and system vulnerabilities [27]. Consequently, blockchain is proposed as a secure platform for storing and sharing data. In some research, blockchain technology has been introduced as a solution for secure and trustworthy exchange of IoT data [28]-[36]. Several researchers have proposed various architectures for secure data storage [32], [33]. In other work, secure data transmission and data management among nodes are guaranteed by blockchain [28]. Other researchers have proposed a patient-centric healthcare data management system [29]. Moreover, some researchers have presented blockchain as a distributed system for data storage with privacy preservation [29], employing pseudonymization through cryptographic functions to maintain patient data confidentiality on the blockchain [29]. Researchers have enhanced the performance of blockchain-based healthcare systems by leveraging wearable sensors and IoT devices [30], [31]. Healthcare data management systems face challenges including data transparency, traceability, immutability, data origin, accessibility, reliability, and privacy. Consequently, a set of transparent concepts and policies has been proposed to

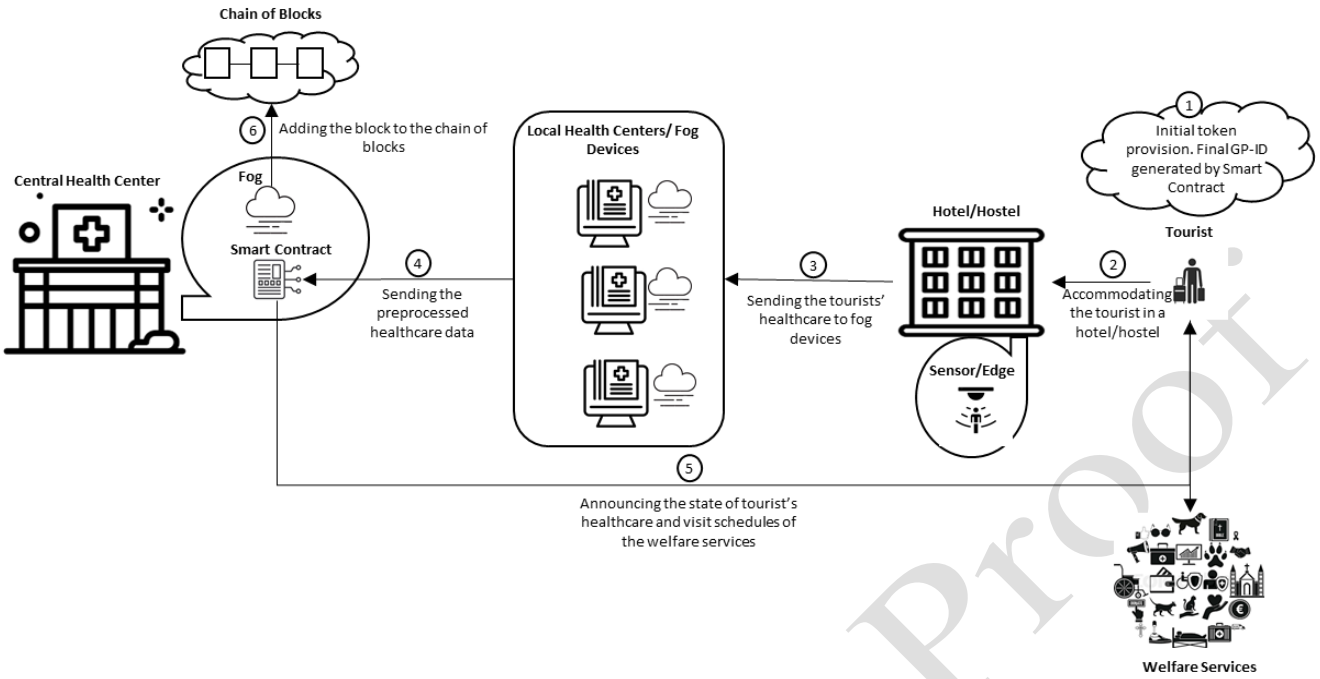

Fig. 3. Components of STHChain

guide the accountability of blockchain-based systems [37]. Other researchers have improved integrity, scalability, and reliability through blockchain technology [27], [38], [39], [49]-[50]. A lightweight, scalable, secure, auditable, and privacy-preserving blockchain-based architecture for healthcare data management has been introduced [27], [31]. The fusion of blockchain and IoT has been recognized as a powerful approach for new business models and distributed applications [40]. Blockchain offers solutions to access-control, privacy, and data-security challenges, while scalability remains a key issue. Studies [41]-[47] employ smart contracts to govern access to healthcare data.

Accordingly, in light of previous work, we advocate a hybrid approach that combines smart city concepts, blockchain, and smart contracts to introduce a model for preserving the smart tourists' healthcare.

2-3. Critical Analysis and Identified Research Gaps

The reviewed literature reveals significant yet often isolated advancements. Research on tourism challenges effectively outlines socioeconomic and health related problems but typically proposes policy or managerial solutions without providing a scalable technical architecture for real-time health monitoring. Concurrently, IoT-based healthcare systems offer sophisticated devices and data pipelines, and blockchain integration addresses security and trust deficits in these systems. However, a critical synthesis exposes three persistent gaps: (1) a lack of integrated frameworks that simultaneously address the specific operational needs of the tourism industry and the technical requirements for secure healthcare IoT, (2) prevalent scalability and energy efficiency bottlenecks in blockchain designs when applied to dense, city-scale sensor networks, and (3) insufficient privacy-preserving identity management tailored for transient populations like tourists, where conventional registration is impractical. The proposed STHChain model is architected explicitly to bridge these gaps. Table 1 synthesizes the identified limitations in prior work and maps them to the specific architectural and protocol-level solutions introduced by STHChain. This structured comparison underscores the model's targeted contributions to scalable, secure, and privacy-aware tourist healthcare.

Table 1. Research Gaps and STHChain's Contributions

Research Area and Limitation in Prior Work	**How STHChain Addresses the Gap**
Tourism Industry Challenges • Proposed solutions are often policy-based, lacking a deployable technical infrastructure for real-time, city-scale health monitoring.	• Provides a hierarchical IoT-edge-fog technical architecture specifically designed for the distributed, dynamic environment of smart tourism. • introduce a formal RBAC model and a transparent, algorithm-based health status function, moving beyond procedural access control descriptions
IoT-based Healthcare Systems • Centralized or cloud-centric models face latency, privacy, and single-point-of-failure issues. • Data integrity and auditability in multi-stakeholder environments are challenging.	• Employs a distributed fog computing layer for low-latency processing and resilience. • Leverages blockchain as an immutable ledger to ensure integrity and enable audit trails for all health data transactions.
Blockchain for Healthcare • Conventional consensus (PoW, PoS) is energy-intensive and slow for high-frequency IoT data. • Storing personal health identifiers on-chain creates severe privacy violations. • Scalability under load from thousands of devices is a common weakness.	• Introduces the lightweight RaC consensus protocol, optimizing energy use and confirmation time via fog node clustering. • Implements a two-tier pseudonymization scheme to decouple patient identity from on-chain data, ensuring privacy-by-design. • The clustered, hierarchical design inherently supports horizontal scaling by adding new fog clusters to handle increased load.

3. Proposed STHChain Model

In this paper, a lightweight and scalable blockchain-based healthcare model called STHChain is proposed for smart tourists, where their access to welfare services is facilitated. The objectives of the model are not only to enhance the blockchain network scalability and reduce energy consumption but also to provide welfare services to tourists in consideration of health protocols during the outbreak of infectious diseases. The model comprises tourists, welfare services, sensors, edge computing, fog computing, and smart contracts, whose interconnections are depicted in Figure 3.

As illustrated in Figure 3, smart hotels/hostels located in smart neighborhoods are equipped with IoT devices. Tourists' vital signs are measured by sensors and collected at edge devices located within the hotels/ hostels. In addition, health centers in each smart neighborhood are equipped with a fog node. Hotels/ hostels then transmit the relevant healthcare data to the fog nodes within the same smart neighborhood for pre-processing. Separately, a smart contract residing at the central health center of the smart city processes the data received from the fog nodes of the health centers in the smart neighborhoods. Based on tourists' healthcare data, this smart contract determines their eligibility to access welfare services during disease outbreaks. Finally, tourists' healthcare data are stored in blocks and recorded on the chain of blocks. The following sections present details of the proposed model.

3-1. Hierarchical Configuration of the IoT-based Healthcare Network

This section describes a hierarchical IoT-based healthcare network for smart tourism. First, hotels/hostels are equipped with sensors and edge devices. For each tourist, a set of sensors is assigned to record the tourist's vital signs in the room where they are staying. The sensors in each room are connected to a specific pin on the edge device, whereby each pin identifies which room's healthcare data is being received. A fog node is then deployed within each hotel/hostel to collect data from the edge devices present there. Each edge device in the hotel/hostel is initially authenticated by this fog node. Subsequently, a fog node is placed at the health center of each neighborhood to perform pre-processing of the healthcare data from the edge devices serving that neighborhood. At the neighborhood health center, the healthcare data of tourists are pre-processed daily. For each tourist, a temporary authentication token is initially provided during hotel check-in. This token is used to generate a secure Local Pseudonym (LP-ID) at the hotel's fog node. This LP-ID is then sent to the central smart contract, which issues a unique Global Pseudonym (GP-ID). The mean sensor data for each tourist is irrevocably associated with this GP-ID on the blockchain, as formalized in Equation (1).

$$TX_i = AvgData_{i,j} \mid GP - ID_j \tag{1}$$

As shown in Equation (1), the transaction associated with sensor *i*, denoted by TX_i, presents the mean data from sensor *i* for tourist *j* as $AvgData_{i,j}$ and the tourist's global pseudonym identifier as $GP - ID_j$. This data is then included in the matrix-Merkle block structure proposed in [36], which organizes transactions into a fixed-size matrix before computing a Merkle root for efficient batch verification. This design ensures data integrity while minimizing storage overhead, a critical consideration for resource-constrained fog nodes.

3-2. Proposed Role-Based Access Control Mechanism and Health Status Determination

The access control in STHChain is governed by a smart contract implementing a formal Role-Based Access Control (RBAC) model. The model is defined by the following core sets and rules:

- ***Sets*****:** *U* (Users: *Tourists, Welfare Centers*), *R* (Roles: *Tourist, ServiceProvider*), *P* (Permissions: *read, write, requestAccess*), *S* (Sessions).
- ***Assignment Rules:*** A user $u \in U$ is assigned a role $r \in R$ by the smart contract based on authenticated credentials. Tourists receive the *Tourist* role upon registration, granting them *read* permission to their own data and *requestAccess* to services. Welfare centers are assigned the *ServiceProvider* role, granting them *write* permission to publish available time slots and conditional *read* permission to verify a tourist's status before granting access.
- ***Conflict Resolution:*** The *ServiceProvider* role inherently overrides the *Tourist* role in defining service parameters. However, access to a specific tourist's data for verification requires a session where the tourist's *requestAccess* permission is actively invoked, ensuring a two-party handshake.

Health Status Determination**:** The classification of a tourist as "healthy" or "unhealthy" is not arbitrary but is determined by a deterministic function executed by the smart contract. Let $AvgData_{i,j}$ represent the mean value from sensor *i* for tourist *j* over a evaluation window (e.g., 24 hours), and let τ_i be a predefined clinical threshold for that vital sign. The health status HS_j for tourist *j* is computed as:

$$HS_j = \begin{cases} \text{'healthy'}, & \text{if } \forall i, |AvgData_{i,j} - \text{Baseline}_i| \leq \tau_i \\ \text{'unhealthy'}, & \text{otherwise} \end{cases}$$

The $Baseline_i$ and thresholds τ_i are set by health authorities and encoded in the smart contract logic. This status is recalculated periodically with new data.

Interaction Flow: The end-to-end process for service access operates through a sequenced dialogue between the entities. It commences when the aggregated health data (*AvgData*) of a tourist is transmitted to the smart contract. The contract subsequently executes its embedded decision function to compute the health status (HS_j) for that tourist and updates the corresponding record on-chain. Independently, welfare services publish their available time slots (*SlotInfo*) to the contract. A tourist can then query the smart contract to retrieve a list of accessible slots, which are dynamically filtered based on their

current HS_j. Upon selecting a suitable slot, the tourist invokes a *requestAccess* function, prompting the contract to generate and issue a temporary, cryptographically-secure access token. This token is presented by the tourist to the welfare service at the point of visit. The service, in turn, submits a verification query to the smart contract, validating both the authenticity of the token and the consistency of the tourist's health status. Only after receiving a positive verification from the contract does the welfare service grant physical access. The entire transaction, culminating in the granted visit, is immutably logged as a new entry on the blockchain, completing the auditable cycle.

3-3. Proposed Raft-based Clustering Consensus Protocol

In this section a Raft-based Clustering consensus protocol called RaC is proposed that employs fog node clustering for verifying tourist healthcare transactions, as illustrated in Figure 4. The approach aims to enhance scalability and energy efficiency of blockchain-based healthcare data storage in smart tourism.

As shown in Figure 4, the smart contract sends tourists' healthcare data and the visitation statistics of welfare centers to the fog node located at the central health center for storing on the chain of blocks. The central health center includes the tourist transactions in blocks and forwards them to the fog nodes situated at local health centers for verification. The fog nodes at nearby local health centers form clusters, and then, using the Raft algorithm [48], perform block verification.

3-3-1. Cluster Formation

Fog nodes are statically grouped into clusters based on their logical affiliation to a specific health center within a smart neighborhood. This grouping is predefined by the system administrator to optimize geographical proximity and network latency. Each cluster consists of a small, fixed set of nodes (e.g., 3 to 5) to maintain Raft's efficiency.

3-3-2. Intra-Cluster Consensus via Raft

Within each cluster, the standard Raft algorithm [48] is executed. One node is elected as the cluster leader, responsible for receiving new block proposals from the central smart contract, appending them to its log, and replicating them to the other follower nodes in the cluster. A log entry (block) is considered committed within the cluster once it is persisted on a majority of the cluster's nodes. This phase ensures data consistency and fault tolerance within the local group.

3-3-3. Cross-Cluster Verification

This phase provides the overarching consensus for the entire network. When a block is committed within a cluster, its leader broadcasts the block digest (hash) to the leaders of all other clusters. Each receiving leader verifies the block against its own log. A block is considered globally verified only after it receives positive verification (VOTE) from a strict majority of cluster leaders (e.g., >50%). This step prevents any single cluster from controlling the ledger.

3-3-4. Final Commitment

Upon achieving global verification, the original cluster leader disseminates a final COMMIT message. All nodes in all clusters then permanently append the verified block to their local copy of the blockchain.

Ultimately, the validated block is added to the chain of blocks in the distributed nodes. The use of fog nodes for block verification not only reduces system load and improves scalability, but also further reduces the energy consumption of the consensus process. Therefore, the tourism industry, by leveraging healthcare data securely stored using blockchain, can better detect tourism patterns during pandemics and leverage these insights for attracting visitors. The detailed step-by-step algorithmic procedure of the RaC protocol is formally presented in Algorithm 1 (see AppendixA).

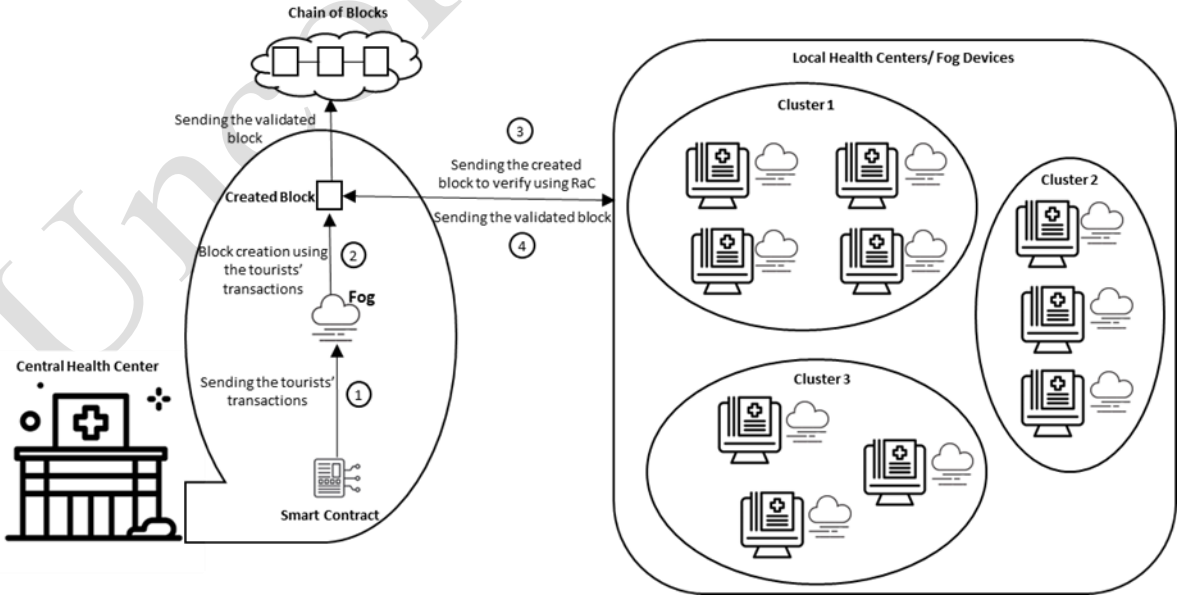

Fig. 4 The proposed RaC Consensus protocol.

3-4. Security Analysis of the RaC Protocol

The RaC protocol inherits and extends security properties from the Raft algorithm and its cross-cluster design to defend against common threats.

- Resilience to Sybil Attacks: In a Sybil attack, an adversary creates multiple fake identities to subvert the consensus. RaC mitigates this by requiring that every fog node is a pre-registered and authenticated entity at a known health center, authorized by the central smart contract. This permissioned setting makes the creation of arbitrary fake nodes practically infeasible.
- Robustness against Consensus Delay Attacks: An attacker might try to deliberately delay consensus. The deterministic leader election and log replication in Raft rely on timeouts; if a leader fails, followers initiate a new election promptly. Furthermore, the cross-cluster verification requires a majority of different clusters. Therefore, compromising or delaying a single cluster (or its leader) cannot halt the entire network's progress, as other clusters can continue to propose and verify blocks.

As demonstrated in lines 14–18 of Algorithm 1, a block is only committed locally after receiving acknowledgements from a majority of its cluster members. This inherent property of the Raft algorithm, combined with the cross-cluster verification (lines 21–42), ensures that compromising a single leader or cluster cannot subvert the consensus, thereby mitigating the risk of a targeted consensus delay attack.

The immutability guaranteed by the RaC protocol directly secures the integrity of the health data and the resulting health status verdict (HS_j), making the access control foundation tamper-evident and trustworthy.

3-5. Privacy-Preserving Identity Management Scheme

Acknowledging the critical privacy concerns associated with using direct personal identifiers, the STHChain model implements a two-tier pseudonymization scheme designed with privacy-by-design principles. This approach ensures that no plaintext, government-issued identifiers (such as passport numbers) are stored on the blockchain or transmitted across the network.

- ***Initial Registration and Local Pseudonym (LP-ID):*** A tourist's passport number is used a single time during the initial check-in at the hotel for verification purposes. Immediately after verification, the system generates a Local Pseudonym (LP-ID). This LP-ID is created by applying a secure cryptographic hash function SHA-256 (We selected SHA-256 for its collision-resistance and widespread adoption in secure systems) to a concatenated string containing the passport number, a random nonce, and a unique salt specific to the hotel. The original passport data is then purged from the operational system. The LP-ID serves as the tourist's anonymous identifier for all local data processing within the hotel's edge and fog layers.
- ***Global Pseudonym (GP-ID) for Blockchain Records:*** When aggregated health data needs to be immutably recorded on the blockchain for access by authorized health centers, a second layer of pseudonymization is applied. The local fog node sends a request containing the LP-ID to a dedicated, central Identity Management Smart Contract. This trusted contract acts as a pseudo-Identity Provider, generating a unique, non-reversible Global Pseudonym (GP-ID) for that specific tourist session. It maintains a secure, encrypted, and access-controlled mapping between the LP-ID and the GP-ID. Only this contract can, under strict conditions defined in its logic (e.g., a legitimate medical emergency request from an authorized health center fog node), resolve a GP-ID back to its corresponding LP-ID, and never to the original passport number. The GP-ID is the sole identifier appended to health transactions written to the blockchain, as shown in the revised Equation (1).

This scheme effectively breaks the linkability between the tourist's real-world identity, their local activity within a hotel, and their global health record on the blockchain. It aligns with data protection principles like minimization and purpose limitation, providing a robust technical safeguard for tourist privacy.

4. Evaluation and Results

This section analyzes the security criteria and evaluates the performance of the proposed model.

4-1. Security Criteria

The proposed model presents the following security criteria using a smart city, blockchain, and smart contracts:

- Authentication: IoT devices in each hotel/hostel are authenticated by the hotel/hostel's own fog nodes. Subsequently, fog devices of hotels/hostels are authenticated by the fog node at the corresponding health center. Additionally, fog nodes located at health centers and the tourists themselves are authenticated by a smart contract.
- Authorization: Tourists are granted read permission, and welfare centers have read/write permissions within the smart contract. Furthermore, healthcare data in this model are retrieved exclusively from authenticated IoT devices, ensuring that only authorized devices may access the network.
- Remote monitoring of healthcare data: IoT devices deployed in hotels/ hostels remotely monitor tourists' vital signs.
- Healthcare data traceability: Tourists' healthcare transactions are traceable in the blockchain using their passport numbers as identifiers.
- Healthcare data integrity: Healthcare data for tourists

are stored as transactions in blocks, which are chained to one another using hashes to ensure integrity.

- Formal access control: A formal Role-Based Access Control model governs all permissions. Access to services is contingent on a deterministic health status verdict computed by the smart contract, ensuring policy enforcement is automated and auditable.
- Data Stewardship and Compliance: The Ministry of Health acts as the data controller within the national framework, responsible for implementing the privacy-preserving architecture and ensuring regulatory compliance. Tourist rights over their personal data are preserved through the system's governance mechanisms.
- Scalability: Fog nodes located at health centers are clustered and, using the proposed RaC protocol, participate in block verification to improve consensus scalability. Additionally, the hierarchical network configuration further enhances the model's scalability.
- Privacy Preserving Identification: Tourist identity is protected through a two-tier pseudonymization scheme. Direct identifiers are never stored. Instead, unlinkable local (LP-ID) and global (GP-ID) pseudonyms are used for processing and blockchain storage, respectively. This design implements a key technical measure recommended by data protection frameworks like the GDPR to mitigate re-identification risks.

4-2 Performance Analysis

The energy consumption, latency, and throughput of the proposed model has been evaluated using the iFogSim simulator. The simulation environment has been configured with hardware specifications of 8 GB RAM and a Core i5 processor. The key entities and network parameters configured in iFogSim to emulate the smart tourism healthcare scenario are comprehensively detailed in Table 2.

4-2-1. Energy Consumption Analysis

The energy consumption of the proposed model is analyzed for the number of sensors, edge devices, and fog nodes, as illustrated in Figure 5.

As shown in Figure 5, energy consumption is measured for scenarios with 25, 50, 100, 150 and 200 sensors connected to four distinct edge devices across four hotels/hostels. The energy consumption is higher when sensor density is low and the sensors are sparsely distributed, compared to higher-density deployments. This is because routing energy for sensors that are more widely spaced incurs greater energy expenditure. However, energy consumption increases gradually with increasing sensor density. Thus, a trade-off between sensor density and energy consumption is necessary. Notably, the energy consumption per sensor decreases as density increases, demonstrating the scalability of the edge-fog preprocessing layer.

Table 2: Summary of Key iFogSim Simulation Parameters

Entity / Parameter	Value / Configuration
Sensor Node	MIPS: 500; Uplink Bandwidth: 512 Kbps
Edge Device (per Hotel)	MIPS: 5000; RAM: 2 GB; Uplink Bandwidth: 10 Mbps
Fog Node (Health Center Cluster)	MIPS: 10000; RAM: 4 GB; Uplink Bandwidth: 100 Mbps
Cloud (Central Health Center)	MIPS: 100000; RAM: 8 GB; Uplink Bandwidth: 1 Gbps
Network Latency (Sensor-Edge)	2 ms
Network Latency (Edge-Fog, Intra-cluster)	5 ms
Network Latency (Fog-Fog, Inter-cluster)	50 ms
Network Latency (Fog-Cloud)	100 ms
Sensor Data Generation	Interval: 5 seconds; Packet Size: 250 bytes
Consensus Timeout (RaC Protocol)	1500 ms

In addition, the energy consumption of the proposed RaC consensus protocol is presented for four clusters, each containing a different number of fog nodes, as shown in Figure 6.

As observed in Figure 6, the energy consumption of the proposed protocol increases with the number of fog nodes in each cluster. While increasing the number of fog nodes in a cluster leads to higher energy usage, it also enhances system reliability because more nodes participate in block verification.

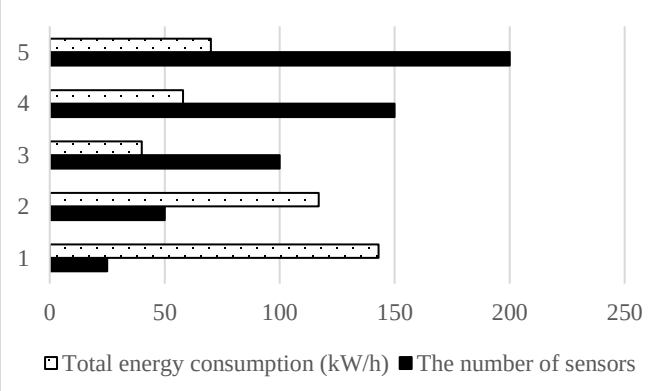

Fig. 5. Energy Consumption of STHChain

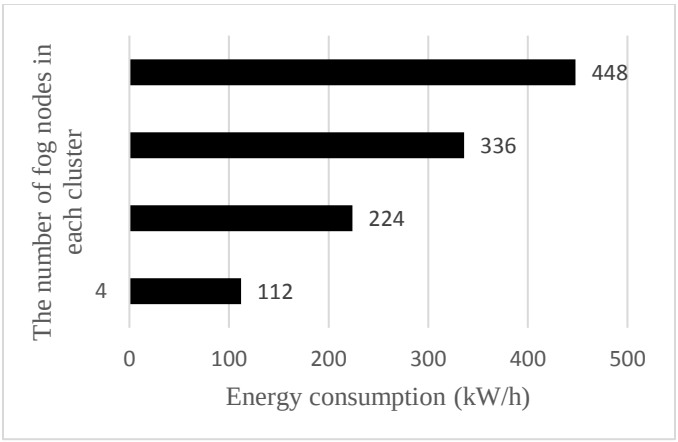

Fig. 6. Energy consumption of RaC for four clusters.

4-2-2. Analysis of Latency and Throughput

Figure 7 illustrates the relationship between transaction load and the average end-to-end latency in the STHChain model. As shown, latency remains below 200 ms for loads up to 750 transactions per second, which is adequate for most real-time health monitoring services. The gradual increase in latency is attributed to queueing overhead in the intermediate Fog nodes, not a central bottleneck. This mild linear behavior indicates the suitable scalability of the architecture.

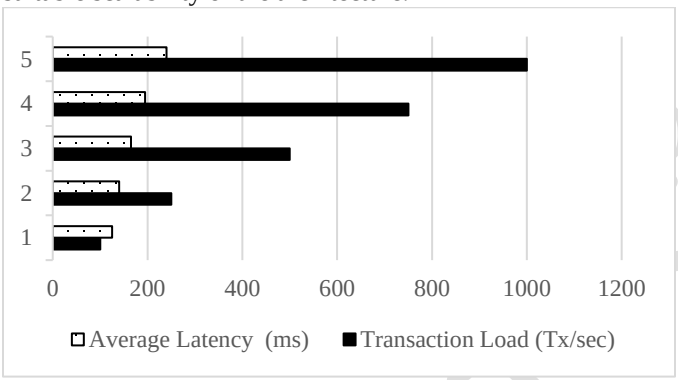

Fig. 7. End-to-End Latency vs. Transaction Load

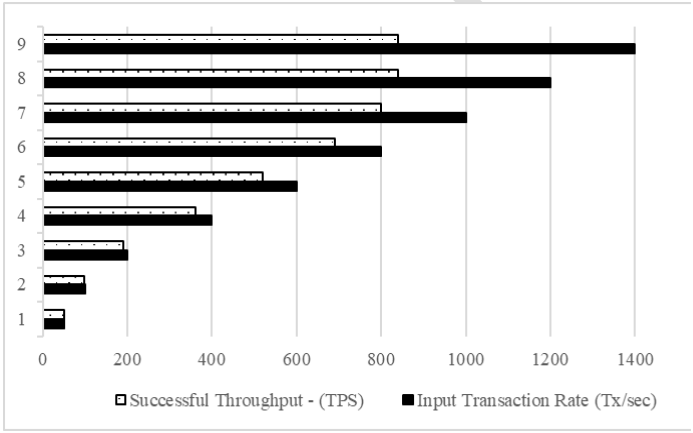

Fig. 8. The system throughput of STHChain (Successful TPS).

The system throughput is presented in Figure 8. The proposed model can process up to approximately 850 transactions per second with a high success rate (>98%). Beyond this point, the throughput saturates due to the inherent bandwidth limitation of the communication link between the Edge and Fog layers. This capacity is fully sufficient for the smart tourism scenario, assuming health data updates from each tourist every few minutes.

4-2-3. Large-Scale Scalability Analysis

In response to the need for evaluating the model under conditions that better reflect expansive smart city deployments, we conducted extended simulations scaling the system up to 500 sensor devices and 8 fog clusters. The objective was to stress-test the architecture and observe performance degradation trends. The results, consolidated in Figure 9, demonstrate the system's scalability limits.

As shown in Fig. 9a, the average end-to-end latency increases in a sub-linear manner with the number of sensors, reaching approximately 580 ms at the maximum scale of 500 sensors. This significant rise is a direct consequence of the configured processing limits (MIPS) at the edge and fog layers, which lead to longer queueing delays under heavy load. While this latency may be acceptable for non-critical health data logging, it underscores the need for provisioning adequate hardware resources for real-time monitoring scenarios. Fig. 9b reveals that the system throughput scales effectively as more fog clusters are added, confirming the benefits of our parallel, clustered design. However, the throughput gain diminishes sharply beyond 6 clusters, with minimal improvement observed at 8 clusters. This indicates that the simulation host's physical memory (8 GB RAM) becomes a bottleneck when orchestrating a large number of virtual fog nodes, highlighting a practical deployment consideration for very large-scale instances of the STHChain architecture.

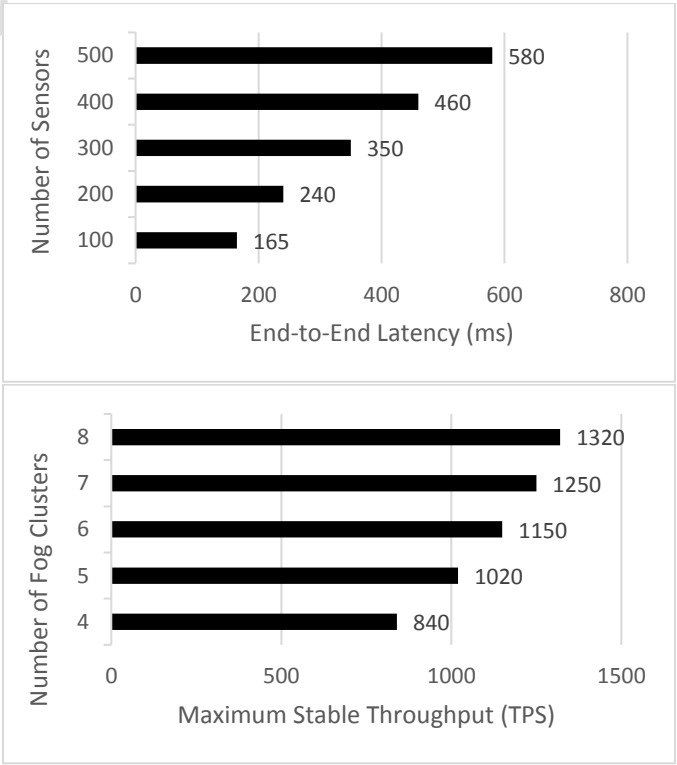

Fig. 9. Large-Scale Scalability Analysis of STHChain- a. Latency vs. Network Size- b. Throughput vs. Cluster Scaling

Table 3. performance comparison of the proposed STHChain model with contemporary blockchain-based frameworks.

Performance Metric	STHChain	Model for Comparison	Explanation and Analysis
Average Latency (End-to-End)	165 ms	~2.8 seconds [38]	STHChain achieves an order-of-magnitude lower latency. This is a direct result of its two-tier fog architecture and RaC protocol, which localizes consensus within hotel/health center clusters, eliminating the wide-area network delays inherent in the compared hierarchical model.
Maximum Throughput (Stable TPS)	840 TPS	~95 TPS [38]	The proposed model demonstrates approximately 9x higher throughput. This scalability stems from the parallel processing capability of independent RaC clusters at different neighborhood health centers, unlike the more sequential verification process described in [38].
Transaction/Block Confirmation Time	1.8 seconds (For RaC consensus)	~0.28 seconds [46]	While the framework in [46] reports faster *transaction time*, it primarily focuses on private data access control on a permissioned blockchain. STHChain's slightly higher time encompasses full block formation and immutable ledger storage for audit trails, providing a stronger data integrity guarantee for medical records.
Energy Efficiency	Optimal (40 units at 100 sensors, Fig. 5)	A Key Challenge [27]	The hierarchical data filtering in STHChain (Edge → Fog → Blockchain) directly addresses the high energy consumption challenge outlined in [27] for lightweight blockchains. By processing and compressing data at the edge, only critical transactions are broadcasted, drastically reducing the network's energy footprint.
Architectural Scalability	Excellent (Horizontal, via fog cluster addition)	Good (Vertical, via node addition in hierarchy [38])	STHChain's design enables linear scalability by adding new independent fog clusters for new city districts. In contrast, scaling the model in [38] requires adding more nodes to the existing hierarchy, which may increase internal communication overhead.

4-3 Comparative Analysis

This section places the performance results of the proposed STHChain model within the broader context of current research by presenting a direct comparison with two contemporary and architecturally relevant blockchain frameworks for healthcare. The objective of this analysis is to validate the efficacy of our design choices specifically, the hierarchical edge-fog configuration and the novel RaC consensus protocol by measuring them against established alternatives. The selected models for comparison are:

A Hierarchical Blockchain Model focused on healthcare scalability [38], which provides a valuable baseline for evaluating structural efficiency in managing distributed health data.

A Privacy-Preserving Access Control Framework for Electronic Health Records (EHR) [46], representing a state-of-the-art solution that prioritizes data security and transaction authorization.

The comparison in Table 3 examines key operational metrics critical for IoT-based health monitoring in smart tourism: end-to-end latency, system throughput, block confirmation time, and inherent architectural scalability. The results demonstrate that STHChain achieves a superior balance across these parameters. Notably, the model's ability to leverage localized consensus via the RaC protocol allows it to overcome the typical trade-off between transaction speed and network scalability observed in other architectures. This positions STHChain as a tailored and efficient solution for real-time, large-scale health data management in dynamic environments like smart cities.

5. Conclusion

In this paper, a lightweight and scalable model for a blockchain-based smart tourist health system called STHChain is proposed to deliver secure and privacy-compliant welfare services for tourists. The model leverages a hierarchical IoT-edge-fog network to optimize energy efficiency and data

processing latency. A core innovation is the formal Role-Based Access Control (RBAC) mechanism and an automated, data-driven health status function implemented within a smart contract. This contract dynamically governs access permissions for tourists, health centers, and welfare services based on objectively evaluated health data, ensuring that service provision is both secure and context-aware.

Tourists' healthcare data, pseudonymized via a two-tier privacy-preserving scheme, and their service access logs are immutably stored on the blockchain. Data integrity and consensus are maintained through the novel Raft-based Clustering (RaC) protocol, which enhances scalability and reduces energy consumption by organizing fog nodes into efficient verification clusters. A comprehensive security analysis confirms the model's resilience against common threats, including Sybil and consensus delay attacks.

The evaluation, conducted using the iFogSim simulator with detailed parameters, demonstrates that STHChain achieves significant gains in energy efficiency, latency, throughput, and scalability under large-scale conditions, outperforming relevant baseline architectures in key metrics for IoT-healthcare systems. Future work will focus on three directions: (1) integrating lightweight machine learning models into the smart contract for real-time health anomaly detection, (2) testing the system in a physical testbed with real IoT devices to validate simulation results, and (3) extending the privacy scheme to support zero-knowledge proofs for selective data disclosure without compromising pseudonymity.

6. References

[1] A. B. Haque, B. Bhushan, and G. Dhiman, "Conceptualizing smart city applications: Requirements, architecture, security issues, and emerging trends," Expert Systems, vol. 39, no. 5, p. e12753, 2022.

[2] N. J. Habeeb and S. T. Weli, "Relationship of smart cities and smart tourism: an overview," HighTech and Innovation Journal, vol. 1, no. 4, pp. 194-202, 2020.

[3] C. Mertzanis and A. Papastathopoulos, "Epidemiological susceptibility risk and tourist flows around the world," Annals of Tourism Research, vol. 86, p. 103095, 2021.

[4] B. Farahani, F. Firouzi, and K. Chakrabarty, "Healthcare iot," in Intelligent internet of things: Springer, 2020, pp. 515-545.

[5] R. Karthick, R. Ramkumar, M. Akram, and M. V. Kumar, "Overcome the challenges in bio-medical instruments using IOT–A review," Materials Today: Proceedings, vol. 45, pp. 1614-1619, 2021.

[6] M. Peyroteo, I. A. Ferreira, L. B. Elvas, J. C. Ferreira, and L. V. Lapão, "Remote monitoring systems for patients with chronic diseases in primary health care: systematic review," JMIR mHealth and uHealth, vol. 9, no. 12, p. e28285, 2021.

[7] A. S. Albahri et al., "IoT-based telemedicine for disease prevention and health promotion: State-of-the-Art," Journal of Network and Computer Applications, vol. 173, p. 102873, 2021.

[8] H. Attaran, N. Kheibari, and D. Bahrepour, "Toward integrated smart city: a new model for implementation and design challenges," GeoJournal, pp. 1-16, 2022.

[9] D. J. Cook, G. Duncan, G. Sprint, and R. L. Fritz, "Using smart city technology to make healthcare smarter," Proceedings of the IEEE, vol. 106, no. 4, pp. 708-722, 2018.

[10] M. Jabbar, K. Prasad, and R. Aluvalu, "Reimagining the Indian Healthcare Ecosystem with AI for a Healthy Smart City," Emerging Technologies in Data Mining and Information Security, pp. 543-551, 2021.

[11] L. Qi et al., "Privacy-aware data fusion and prediction with spatial-temporal context for smart city industrial environment," IEEE Transactions on Industrial Informatics, vol. 17, no. 6, pp. 4159-4167, 2020.

[12] M. Gheisari et al., "OBPP: An ontology-based framework for privacy-preserving in IoT-based smart city," Future Generation Computer Systems, vol. 123, pp. 1-13, 2021.

[13] J. Venkatesh, B. Aksanli, C. S. Chan, A. S. Akyurek, and T. S. Rosing, "Modular and personalized smart health application design in a smart city environment," IEEE Internet of Things Journal, vol. 5, no. 2, pp. 614-623, 2017.

[14] S. S. F. Nasab, T. Z. Marjaneh, and D. Bahrepour, "Energy Efficiency and Establishing Service Level Agreement using Fuzzification of Virtual Machine Selection Policies for Migrating in Cloud Computing," in Proc. 9th Int. Conf. Web Res. (ICWR), pp. 201–207, 2023.

[15] S. S. Fateminasab, S. Memarian, S. R. K. Tabbakh, and M. C. Romero-Ternero, "A Review on Open Data Storage and Retrieval Techniques in Blockchain-based Applications," in 2024 10th International Conference on Web Research (ICWR), 2024: IEEE, pp. 297-302.

[16] S. S. F. Nasab, D. Bahrepour, and S. R. K. Tabbakh, "A Review on Secure Data Storage and Data Sharing Technics in Blockchain-based IoT Healthcare Systems," in 2022 12th International Conference on Computer and Knowledge Engineering (ICCKE), 2022: IEEE, pp. 428-433.

[17] S.-Y. Pan, M. Gao, H. Kim, K. J. Shah, S.-L. Pei, and P.-C. Chiang, "Advances and challenges in sustainable tourism toward a green economy," Science of the total environment, vol. 635, pp. 452-469, 2018.

[18] S. B. Hojeghan and A. N. Esfangareh, "Digital economy and tourism impacts, influences and challenges," Procedia-Social and Behavioral Sciences, vol. 19, pp. 308-316, 2011.

[19] D. Dredge and S. Gyimóthy, "Collaborative economy and tourism," Collaborative Economy and Tourism: Perspectives, Politics, Policies and Prospects, pp. 1-12, 2017.

[20] M. Abubakar et al., "Medical tourism in Nigeria: Challenges and remedies to health care system development," International journal of development and management review, vol. 13, no. 1, 2018.

[21] J. P. Singh, "Healthcare tourism in India: Opportunity and challenges," Asian Journal of Multidimensional Research (AJMR), vol. 4, no. 3, pp. 37-47, 2015.

[22] R. H. Chianeh, G. Del Chiappa, and V. Ghasemi, "Cultural and religious tourism development in Iran: Prospects and challenges," Culture and Cultures in Tourism, pp. 86-95, 2020.

[23] R. Sharpley, "Tourism, modernisation and development on the island of Cyprus: Challenges and policy responses," Journal of sustainable tourism, vol. 11, no. 2-3, pp. 246-265, 2003.

[24] S. P. Singh, A. Nayyar, R. Kumar, and A. Sharma, "Fog computing: from architecture to edge computing and big data processing," The Journal of Supercomputing, vol. 75, pp. 2070-2105, 2019.

[25] Q. Qi and F. Tao, "A smart manufacturing service system based on edge computing, fog computing, and cloud computing," IEEE access, vol. 7, pp. 86769-86777, 2019.

[26] Y. Shi, G. Ding, H. Wang, H. E. Roman, and S. Lu, "The fog computing service for healthcare," in 2015 2nd International Symposium on Future Information and Communication Technologies for Ubiquitous HealthCare (Ubi-HealthTech), 2015: IEEE, pp. 1-5.

[27] L. Ismail, H. Materwala, and S. Zeadally, "Lightweight blockchain for healthcare," IEEE Access, vol. 7, pp. 149935-149951, 2019.

[28] A. Abbas, R. Alroobaea, M. Krichen, S. Rubaiee, S. Vimal, and F. M. Almansour, "Blockchain-assisted secured data management framework for health information analysis based on Internet of Medical Things," Personal and ubiquitous computing, pp. 1-14, 2021.

[29] A. Al Omar, M. S. Rahman, A. Basu, and S. Kiyomoto, "Medibchain: A blockchain based privacy preserving platform for healthcare data," in Security, Privacy, and Anonymity in Computation, Communication, and Storage: SpaCCS 2017 International Workshops, December 12-15, 2017, Proceedings 10, 2017: Springer, pp. 534-543.

[30] M. T. Quasim, F. Algarni, A. A. E. Radwan, and G. M. M. Alshmrani, "A blockchain based secured healthcare framework," in 2020 International Conference on Computational Performance Evaluation (ComPE), 2020: IEEE, pp. 386-391.

[31] M. Qi et al., “Privacy protection for blockchain‐based healthcare IoT systems: A survey,” IEEE/CAA J. Autom. Sinica, vol. 11, no. 8, pp. 1757–1776, Aug. 2024..
[32] R. Li, T. Song, B. Mei, H. Li, X. Cheng, and L. Sun, "Blockchain for large-scale internet of things data storage and protection," IEEE Transactions on Services Computing, vol. 12, no. 5, pp. 762-771, 2018.
[33] W. Liang, Y. Fan, K.-C. Li, D. Zhang, and J.-L. Gaudiot, "Secure data storage and recovery in industrial blockchain network environments," IEEE Transactions on Industrial Informatics, vol. 16, no. 10, pp. 6543-6552, 2020.
[34] Y. Y. Yap et al., “Smart tourism technologies and tourist satisfaction: A systematic literature review and research agenda,” Acta Psychol., vol. 258, p. 105191, Aug. 2025.
[35] S. S. Fateminasab, D. Bahrepour, and S. R. K. Tabbakh, "A fair non-collateral consensus protocol based on Merkle tree for hierarchical IoT blockchain," Scientific Reports, vol. 15, no. 1, p. 3645, 2025.
[36] S. S. Fateminasab, N. Evaznia, S. Memarian, M. d. C. Romero-Ternero, G. Miró-Amarante, and S. R. K. Tabbakh, "An efficient blockchain-based resource allocation and secure data storage model using Fire Hawk Optimization and entropy in health tourism," Journal of Cloud Computing, vol. 14, no. 1, p. 63, 2025.
[37] F. Rizal Batubara, J. Ubacht, and M. Janssen, "Unraveling transparency and accountability in blockchain," in Proceedings of the 20th annual international conference on digital government research, 2019, pp. 204-213.
[38] L. Sadath, D. Mehrotra, and A. Kumar, "Scalability Performance Analysis of Blockchain Using Hierarchical Model in Healthcare," Blockchain in Healthcare Today, vol. 7, 2024.
[39] S. R. Mallick, R. K. Lenka, P. K. Tripathy, D. C. Rao, S. Sharma, and N. K. Ray, "A lightweight, secure, and scalable blockchain-fog-iomt healthcare framework with ipfs data storage for healthcare 4.0," SN Computer Science, vol. 5, no. 1, p. 198, 2024.
[40] H. F. Atlam, A. Alenezi, M. O. Alassafi, and G. Wills, "Blockchain with internet of things: Benefits, challenges, and future directions," International Journal of Intelligent Systems and Applications, vol. 10, no. 6, pp. 40-48, 2018.
[41] A. Saini, Q. Zhu, N. Singh, Y. Xiang, L. Gao, and Y. Zhang, "A smart-contract-based access control framework for cloud smart healthcare system," IEEE Internet of Things Journal, vol. 8, no. 7, pp. 5914-5925, 2020.
[42] R. Akkaoui, X. Hei, C. Guo, and W. Cheng, "RBAC-HDE: On the design of a role-based access control with smart contract for healthcare data exchange," in 2019 IEEE International Conference on Consumer Electronics-Taiwan (ICCE-TW), 2019: IEEE, pp. 1-2.
[43] R. Kumar and R. Tripathi, "Scalable and secure access control policy for healthcare system using blockchain and enhanced Bell–LaPadula model," Journal of Ambient Intelligence and Humanized Computing, vol. 12, pp. 2321-2338, 2021.
[44] S. Tanwar, K. Parekh, and R. Evans, "Blockchain-based electronic healthcare record system for healthcare 4.0 applications," Journal of Information Security and Applications, vol. 50, p. 102407, 2020.
[45] N. Kshetri, "Blockchain and electronic healthcare records [cybertrust]," Computer, vol. 51, no. 12, pp. 59-63, 2018.
[46] A. K. Jakhar, M. Singh, R. Sharma, W. Viriyasitavat, G. Dhiman, and S. Goel, "A blockchain-based privacy-preserving and access-control framework for electronic health records management," Multimedia Tools and Applications, pp. 1-35, 2024.
[47] A. Raj and S. Prakash, "An Efficient Blockchain-Based Access Control Framework for IoT-Healthcare System," Wireless Personal Communications, vol. 136, no. 2, pp. 1017-1045, 2024.
[48] D. Ongaro and J. Ousterhout, "The raft consensus algorithm," Lecture Notes CS, vol. 190, p. 2022, 2015.
[49] Soleimani Nadaf, S., Niazi Torshiz, M., Mazinani, S. et al. A lightweight scalable and dynamic blockchain-based model for storing and retrieving patient healthcare records. Sci Rep 15, 44568 (2025).
[50] M. F. A. Mohammed et al., “A cluster‐based architecture for fog computing,” IEEE Access, vol. 13, pp. 187638–187657, 30 Oct. 2025.

AppendixA

Algorithm1 : Raft-based Clustering (RaC) Consensus Protocol

==========

Definitions:

FN[] : Array of all Fog Nodes in the network

CL[] : Array of Clusters. CL[k] is a list of Fog Nodes in clusterk

SmartContract : The central controller

Block : The data block to be appended

majority(n) : Function returning smallest integer > n/2

HealthStatus: The health classification ('healthy'/'unhealthy') for a tourist, computed by the smart contract.

Input: A new Block proposed by the SmartContract

Output: Block appended globally or an error message

PHASE 1: Cluster-Centric Block Proposal

Procedure PROPOSE_BLOCK(Block):

1. primaryCluster = SELECT_PRIMARY_CLUSTER()
   // e.g., cluster nearest to the data source
2. primaryLeader = GET_LEADER(primaryCluster)
3. Send Block from SmartContract to primaryLeader

End Procedure

PHASE 2: Intra-Cluster Consensus (Raft)

Procedure INTRA_CLUSTER_CONSENSUS(Block, leader):

4. leader.log.append(Block)
5. // Replicate block to all followers in the same cluster
6. FOR each follower IN leader.cluster:
7. Send AppendEntriesRPC(Block) to follower
8. END FOR
9. // Wait for acknowledgements from a majority
10. IF acks_received >= majority(size(leader.cluster)) THEN
11. Block.status = COMMITTED_LOCALLY
12. RETURN SUCCESS
13. ELSE
14. RETURN FAILURE // Triggers leader re-election or retry
15. END IF

End Procedure

PHASE 3: Cross-Cluster Verification

Procedure CROSS CLUSTER VERIFICATION (Block, primaryLeader):

16. verificationVotes = 0
17. // Send verification request to other cluster leaders
18. FOR each cluster IN CL WHERE cluster != primaryLeader.cluster:
19. otherLeader = GET_LEADER(cluster)
20. Send VerificationRequest(Block.hash) to otherLeader
21. END FOR
22. // Collect validation votes
23. FOR each otherLeader IN otherLeaders:
24. vote = receive VerificationResponse from otherLeader within TIMEOUT
25. IF vote == VALID THEN
26. verificationVotes = verificationVotes + 1
27. END IF
28. END FOR
29. // Check for inter-cluster majority
30. IF verificationVotes >= majority(size(CL)) THEN
31. Block.status = GLOBALLY_VERIFIED
32. RETURN SUCCESS
33. ELSE
34. RETURN FAILURE // Block is rejected
35. END IF

End Procedure

PHASE 4: Final Commitment & Broadcasting

Procedure FINAL_COMMIT(Block, primaryLeader):

36. // Broadcast final commit order to all nodes in all clusters
37. FOR each node IN FN:

```
    38.     Send FinalCommitOrder(Block) to node
    39.     node.blockchain.append(Block)
    40. END FOR
    41.  Broadcast EVENT: "Block <Block.ID> committed
successfully"
End Procedure
================================================
MAIN EXECUTION FLOW
================================================
Procedure MAIN_RAC(Block):
   // Step 1: Propose block to a primary cluster
   PROPOSE_BLOCK(Block)
   // Step 2: Reach consensus within the primary cluster
   IF  INTRA_CLUSTER_CONSENSUS(Block,  primaryLeader)
== SUCCESS THEN
      // Step 3: Validate block across different clusters
      IF                CROSS_CLUSTER_VERIFICATION(Block,
primaryLeader) == SUCCESS THEN
         // Step 4: Finalize and broadcast the block
         FINAL_COMMIT(Block, primaryLeader)
         RETURN  "Block  committed  successfully  to  the
blockchain."
      ELSE
         RETURN "ERROR: Cross-cluster verification failed. Block
rejected."
      END IF

   ELSE
      RETURN "ERROR: Intra-cluster consensus failed. Block not
verified."
   END IF
End Procedure
================================================
```