Review of Architectures in IoT Remote Patient Monitoring Systems

Lan Sovinc and Marko Bajec

University of Ljubljana,
Večna pot 113, 1000 Ljubljana, Slovenia
ls7312@student.uni-lj.si
marko.bajec@fri.uni-lj.si

Abstract. This review article examines different architectural approaches for IoT-based Remote Patient Monitoring Systems. Four different classification categories have been identified, each with their own strengths and weaknesses.

Fog-based architectures reduce latency and enhance privacy through localized data processing. Cloud-based solutions offer robust storage and analytics capabilities, but struggle with latency and centralization. Hybrid systems combine edge and cloud computing to improve scalability and efficiency, while blockchain-enabled architectures focus on secure, decentralized data management.

The findings from 11 original publications were summarized. This paper describes the characteristics of the identified approaches by examining common and unique concepts used by the solutions in the same classification category based on prior analysis.

Keywords: Remote Patient Monitoring, RPM, Internet of Things, IoT, IoT Healthcare Systems, Fog Computing, Cloud Computing, Blockchain, Scalable RPM Systems, Wearable Devices, Real-time Health Monitoring.

1. Introduction (Architecture approaches for IoT RPM)

Remote Patient Monitoring (RPM) systems are being increasingly discussed in modern healthcare [7]. RPMs relieve patients of manual health data collection and provide more representative measurements as they are being taken in their home environment. On the other hand, they benefit the medical institutions and their staff by reducing the number of physical visits, better monitoring of a higher number of patients and extending the range of medical services to a greater geographical area.

To implement a modern RPM system, different technologies can be chosen. One promising example is the Internet of Things (IoT). IoT and its healthcare-specific subset, IoMT (Internet of Medical Things), are being increasingly recognized for their potential to address medical requirements efficiently, especially in terms of scalability, security, and real-time monitoring [1]. The widespread use of this technology for home automation suggests growing maturity and feasibility of exploring this area for use in environments where the reliability and accuracy of systems and their measurements are critical. However, implementation of such solutions using IoT brings several system design challenges that should be addressed to improve effectiveness [11].

This review focuses on analyzing and classifying the architectural paradigms employed in IoT-based RPM systems examined and how they ensure high availability, patient privacy and scalability. Four distinct classes of IOT RPM system were identified: cloud-based, fog-based, hybrid, and blockchain-enabled. By classifying solutions based on data processing and system functionality, the review examines innovations and tradeoffs in RPM design. This work aims to contribute to understanding the strengths and limitations of architectural approaches, providing valuable insight to system designers and researchers to improve IoT-based RPM systems.

The remainder of this paper is structured as follows. Section 2 (Search strategy) describes the literature search and selection process. Section 3 (Publication Classification) presents the classification framework used to group the identified RPM architectures. Sections 4–7 then discuss fog-based, cloud-based, hybrid, and blockchain-enabled architectures in detail. Finally, the Conclusion summarizes key findings and outlines directions for future research on IoT-based RPM system design.

2. Search strategy

Studies published before 1st February 2024 were retrieved from The IEEE Xplore digital library database. The following search term was used: IoT AND (”Remote Patient Monitoring” OR ”RPM”) AND (”Non-Invasive Technologies” OR ”Digital Health Technologies” OR ”Wearable Devices”). The search period was limited to publications published from 2018 onward in order to restrict the search window to the last five years, calculated from the time we started writing the article. Our search yielded 35 results. The abstracts and full texts of each publication were screened and assessed for inclusion. Out of 35 initial publications, 23 were excluded for several reasons:

- 16 did not discuss the topic of our review article directly
- 6 were reviews without original data
- 1 was a survey
- 1 was a physical-only resource that we did not have access to

Finally, 11 original publications were included in this review.

3. Publication Classification

We analyzed the selected publications in order to develop a system for classifying their discoveries into different categories. In this way, we could later compare the details of the solutions that used the same approaches. By looking only at the specifics of the architecture within a particular paradigm, we sought to enable more direct comparisons.

We decided to classify the solutions according to where most of the data is processed, in relation to the role of the device in the system. With this approach, we formed 4 categories. “Fog-based architecture”, where the processing power is partially transferred from the central server to the periphery, as opposed to the more traditional “cloud-based architecture”, where the powerful processing takes place on a centralized remote server or group of servers. Then we observed some approaches where simple operations are performed on the edge devices, but more complex analytics are moved to the cloud, which we call “hybrid architecture”. Finally, we observed a interest in the use of blockchain paradigms to fulfill the requirements of an IoT RPM system and decided to include a chapter discussing the pros and cons of integrating “blockchain technology” in the system architecture.

All examined solutions, as represented in the supplementary material, are provided for reference.

4. Fog-based Architecture

In the early stages of the development of RPM IoT system architectures, there were two main approaches: Edge-based and cloud-based approaches. Both have their own advantages and disadvantages, which we will discuss in later chapters. For some years now, however, there has been a middle ground between these two approaches, namely the use of the so-called fog computing paradigm. The term was invented to refer to “cloud computing” by comparing how fog appears in nature in relation to clouds. In this case, some of the computational capabilities are spread from a central cloud closer to the IoT devices (the fog). Such devices are usually responsible for communicating with multiple IoT devices on the same local network. A commonly cited advantage of fog nodes is their relative computing power compared to the IoT sensors. This can reduce central server load through local filtering and processing, may enhance the privacy aspect of such systems as less data is sent outside the patient’s home network, and can support more advanced emergency response capabilities.

The first example of a fog-based infrastructure is the one in [5]. The goal of the RPM system was a low-latency approach to detect abnormalities of patients’ heart rhythm by ECG (electrocardiogram) using IoT wearable devices and deep learning. The system is represented by the sensor, fog and cloud layers. Wearable IoT devices are part of the sensor layer, which is capable of transmitting data to the fog layer and responding to commands from the fog layer when medical emergencies are detected. The devices in this layer are the simplest and therefore cannot perform data analysis. The fog layer consists of a cluster of fog devices, called gateways, which act as intermediaries between the sensors and the cloud. Each fog node has its own local storage for the temporary storage of measurement data. If the node cannot process the signal locally due to a high computational load, the data is transferred from the local storage to the cloud for processing. In a scenario without overload, the signal is first filtered and then processed on the node. As they are computationally more powerful than the sensor level devices, but still far less powerful than the cloud, the fog nodes perform the signal classification using the inference model downloaded from the cloud, but do not implement the deep learning process. If an abnormal value is detected, the alert is first sent to the sensor layer to notify the emergency services before being sent to the cloud layer for model training. The cloud layer is responsible for the global storage of the measured values, the constant training process of the deep learning model and the transmission of the latest inference model to the fog layer. The system architecture was tested together with a one-dimensional convolutional neural network as a classification model. The authors reported a 25% faster response time compared to the conventional cloud-based approach.

The second fog-based computing architecture [6] attempts to optimize patient privacy, data processing, system response time and patient-service interaction. The system uses

IoT sensors in combination with a surveillance camera feed and speech recognition devices to improve interaction. The fog node is the main component of the framework. The sensor data is transmitted to the fog device, which processes it and obtains the patient's consent to connect via the voice recognition module if an abnormal value is detected. The medical center is only given permission to view the camera feed if the patient has given consent or if there is an emergency (if the patient is asked for consent but does not respond). In this way, the fog node fulfills three tasks. It provides local data filtering that ensures patient privacy. By checking for anomalies and speech recognition in the fog node, processing is offloaded from the main server. Emergency confirmation process is improved using the camera feed. The system prototype was implemented with 3 Raspberry Pis for speech recognition and an NVIDIA Jetson TX1 as a fog node. Compared to a conventional system architecture, where all processing takes place in the cloud, the fog-enabled system reported an average round-trip latency for processing requests that was 100 times lower.

The gateway does not always have to be a separate device. To save costs and make better use of hospital resources, [9] developed an early warning system that uses the patient's smart device as a gateway. The aim of the system was to detect a sudden deterioration in a patient's health after discharge from the ICU (intensive care unit). The system consists of wearable IoT devices, a smartphone or tablet that serves as a gateway, and a remote early warning and analysis system. The medical data from the IoT devices is transmitted to the gateway via Bluetooth. There, the raw data is processed using specially developed mathematical models and the current health metrics are generated. Communication between the gateway and the remote analysis system takes place via local or cellular networks. The remote analysis system displays the readings to the medical staff, but also provides automatic alerts with an early warning value for the emergency response team. By utilizing existing resources, the team was able to quickly implement the system and conduct a successful pilot study with more than one hundred patients.

The examples of architectures described above show that fog computing paradigms may offer several potential benefits for use in IoT-based Remote Patient Monitoring (RPM) systems. It combines the strengths of edge and cloud computing while mitigating their respective limitations. By decentralizing computational tasks closer to the data sources, fog architectures offer improved data protection, reduced server load and improved emergency response capabilities. Comparing the solutions examined, fog implementations differ in their approach to data protection, latency and resource utilization, despite their common basis in the distribution of processing tasks. For example, the integration of advanced data processing and emergency protocols directly in the fog layer, as seen in [5], contrasts with architectures that prioritize patient privacy through localized consent mechanisms [6]. Furthermore, the innovative use of existing devices as gateways [9] emphasizes the versatility and cost-effectiveness of fog computing in healthcare. Overall, these examples indicate a potential role for fog computing in the development of RPM systems by providing a balanced, adaptable framework that meets the critical needs of healthcare IoT infrastructure.

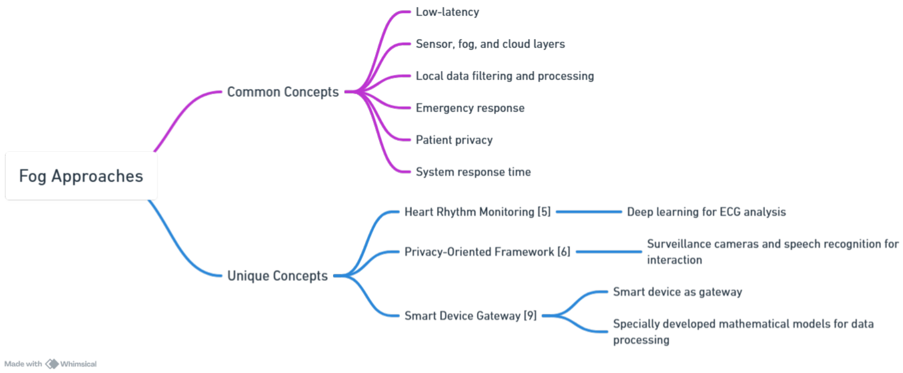
Fog Approaches
Common Concepts
Low-latency
Sensor, fog, and cloud layers
Local data filtering and processing
Emergency response
Patient privacy
System response time
Unique Concepts
Heart Rhythm Monitoring [5]
Deep learning for ECG analysis
Privacy-Oriented Framework [6]
Surveillance cameras and speech recognition for interaction
Smart Device Gateway [9]
Smart device as gateway
Specially developed mathematical models for data processing
Made with Whimsical

Fig. 1. Mind map for Fog-based architectures comparison

5. Cloud-based Architecture

If we want to perform complex data processing, have centralized storage of medical records or use sophisticated data analysis approaches such as machine learning or deep learning, we need powerful resources that a cloud-based approach can provide. In such scenarios, the IoT devices only transmit the data to the cloud and sometimes also receive the analysis results. Such systems can provide substantial processing capacity, but the dependence on a central server may introduce a single point of failure. Another challenge is the latency that is sometimes required for real-time RPM systems. As the amount of data at the ingestion point increases and data processing becomes more complex, the latency can become more pronounced.

The fog node is not required if the sensor devices themselves are equipped with sufficiently powerful hardware. This means that the medical IoT device can be connected directly to the central server, for example via Wi-Fi. This simplifies the architecture, but places a greater load on the central server. By using an established cloud infrastructure provider, the load can be distributed internally.

The smart healthcare monitoring system by [13] used this approach by combining Wi-Fi IoT sensors with a cloud database Firebase. They implemented a real-time monitoring solution with two mobile applications, one for patients and one for medical staff. By developing a background service, alerts were implemented for both parties, even when the mobile application is not actively running on their devices. The Firebase Realtime Database API was used to respond to data changes without the need for user intervention or manual updating of the app, contributing to the real-time requirements of the system.

A similar hardware approach was taken in [8], where a centralized system was developed for people infected during a pandemic. It allows the authorities to verify that the person is following quarantine instructions and the medical staff and family to monitor the patient's health status while providing a standardized EHR (electronic health record) cloud data storage solution. The system consists of three layers. IoT wearable devices take medical measurements and forward them to the online platform along with the device's GPS (Global Positioning System) data. The platform consists of a database and a server on which the data processing takes place. First, the platform checks whether the transmitted vital data is within the normal range and only if this is not the case are the GPS coordinates checked. Based on this data, an emergency alert can be triggered to the authorities if the patient's location does not match the agreed quarantine location. The medical measurement is then standardized as an EHR and written to the cloud database. A web-based monitoring interface has been implemented for medical staff. The server also servers the data to the mobile companion app, which the patient's relatives can use to find out about their current state of health. A limitation of this solution is that the automatic alerts are only implemented for quarantine violations, but not for abnormal medical measurements, which still require manual review by medical staff or the patient's relatives.

IoT Remote Patient Monitoring (RPM) systems that use a cloud-based infrastructure share a certain degree of common characteristics of the models discussed. All of the architectures examined leverage the power of cloud-based data processing and centralized storage, which underpins their robust ability to process complex data operations and enable monitoring capabilities through IoT devices. Despite these common attributes, both solutions focused on different types of functionalities. Article [13], for example, features seamless real-time data synchronization and uses a background alert service to enable seamless monitoring without direct app interaction. On the other hand, Article [8] integrates GPS data to enforce quarantine compliance, providing a unique layer of patient management. It also introduces a web-based interface for healthcare professionals and selectively automates alerts for quarantine violations, while other critical alerts are not automated. These nuanced differences highlight the tailored approaches to address specific healthcare monitoring challenges and emphasize the importance of context-driven architecture design in IoT-based patient care.

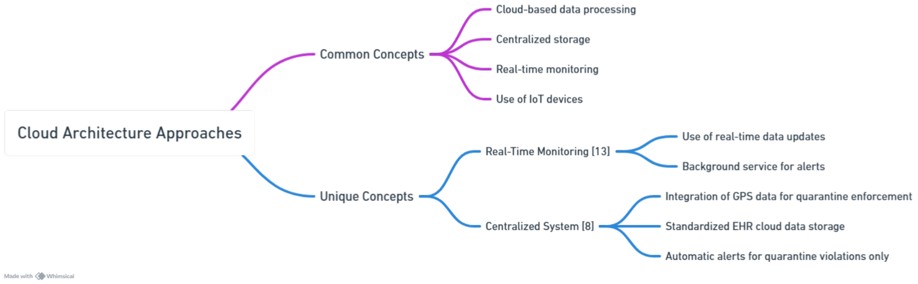
Cloud Architecture Approaches
Common Concepts
Cloud-based data processing
Centralized storage
Real-time monitoring
Use of IoT devices
Unique Concepts
Real-Time Monitoring [13]
Use of real-time data updates
Background service for alerts
Centralized System [8]
Integration of GPS data for quarantine enforcement
Standardized EHR cloud data storage
Automatic alerts for quarantine violations only
Made with Whimsical

Fig. 2. Mind map for Cloud-based architectures comparison

6. Hybrid Architecture

The edge-based solutions are decentralized, straightforward to implement and not dependent on another external system to perform the basic tasks. They usually consist of small battery-powered IoT devices that may or may not communicate with external systems, but perform their data analysis independently. The advantage of these devices is that they are able to run simple algorithms to perform basic local signal analysis. However, due to their size, they are limited in terms of resources and power. This subsection discusses architectural solutions that mainly utilize edge computing paradigms but seek to mitigate limitations of requiring powerful edge devices by utilizing the cloud for more complex tasks and data centralization.

In both solutions [2] and [4], approaches were developed in which a ubiquitous device, e.g. the patient's smartphone, served as a gateway between the IoT devices and the cloud. The main difference to the fog-based solution is that apart from data aggregation and forwarding, no data processing takes place locally on the device itself.

The solution in [2] utilized IoT smart patches to measure ECG signals from patients and developed an RPM framework for monitoring and real-time data analysis using a microservice architecture to achieve better performance and scalability. Data acquisition, streaming, processing, storage and reporting were implemented. To overcome the resource constraints of smart patches, data is transmitted via BLE (Bluetooth Low Energy) to the nearest paired smartphone. An app was developed for this device, which serves as a gateway to the cloud and enables the management of IoT devices, the display of real-time data and alerts for patients. The app also creates a background service that connects to the backend via MQTT (Message Queuing Telemetry Transport) protocol to enable bidirectional communication. The core of the backend consists of the open source IoT platform ThingsBoard. It enables the system to store patient personal data and time series data from the IoT patches, send notifications to the smartphone app and communicate with other backend services via the so-called "integrations". Two main integrations are those for the MQTT broker for data acquisition and for the event storage and stream processing platform. The scalability of the system is achieved through the Apache Kafka event store. It enables the system to forward the data to the corresponding microservice without overloading its resources. It prepares the data and waits until the target service is ready to process it. The key microservice is the one for data analysis with machine learning algorithms, which enables real-time alerts and decision making. After the data is analyzed, a new event is created containing the results of the data classification along with the raw data to be written to the database for long-term storage. This database is also used as a source for the dashboard web application, which displays the ECG charts to the medical staff and informs them of any anomalies in their patients' measurements. An advantage claimed for this approach, which overcomes the traditional disadvantage of cloud architecture, is the use of an event processing platform together with the microservice backend architecture to achieve sustained performance of the system as the amount of data at the entry point increases.

Another case of a hybrid approach was described in [4], in which both the cloud and the resources already available (patient's smartphone) were used. The system implements all the main functions of the larger systems, but by reusing existing resources and developing a single code base, the solution is very compact and therefore easy to maintain and deploy. The aim of the system was to implement RPM, in particular to measure heart rate and oxygen saturation and to provide a a solution intended to be readily available and low-cost. The developed app not only provides the management interface for the system, but can also capture medical data from the device's camera to measure heart rate and blood oxygen saturation using photoplethysmography. The same measurements could also be taken in a more precise way using an IoT sensor device with more specialized sensor equipment. In this case, the IoT device transmits all data to the cloud independently, and the smartphone is only used to display the data by querying the server. The same app is used for patients and doctors. Patients can view their measurement data, receive medical alerts and manage medical services. Doctors can view their patients' data and communicate with them via the integrated chat function.

Hybrid architectural approaches in IoT Remote Patient Monitoring (RPM) systems show a mix of edge and cloud computing paradigms to leverage the strengths of both systems while mitigating their respective weaknesses. Commonalities between the solutions discussed include the use of patient-owned devices as intermediary gateways, system scalability and ensuring user-friendly interfaces. Differences can be seen in the scope of data processing, with some approaches favoring minimal edge computation and others incorporating extensive cloud-based analytics and decision-making capabilities. Of note is the innovative use of existing resources (e.g., smartphones) to reduce system complexity and cost, as in solution [4], in contrast to the more elaborate service-oriented architecture of solution [2], which emphasizes performance and scalability through microservices and event-driven design. These approaches demonstrate the critical balance between resource efficiency, real-time responsiveness, and system scalability in designing effective remote patient monitoring (RPM) systems.

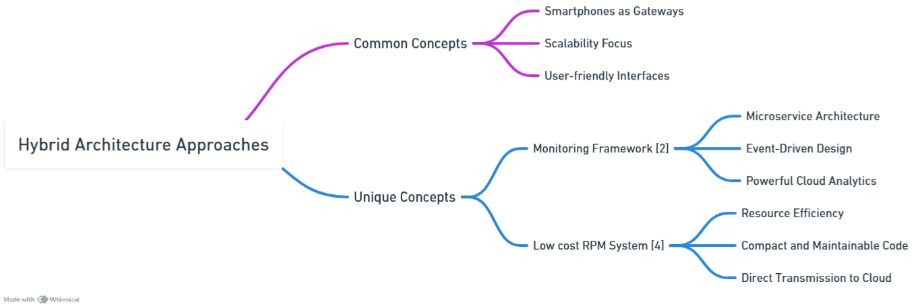

Fig. 3. Mind map for Hybrid architectures comparison

7. Blockchain Technology in RPM

Blockchain technologies have seen broader adoption beyond digital currencies. Several articles that dealt specifically with how utilizing blockchain could improve the current RPM system architectural approaches. Even though, the described solutions utilize the paradigms of all previously described architectures, we decided to include it into a separate section as they still act distinctly differently from the traditional systems.

This section discusses potential benefits of blockchain technologies to IoT RPM systems, how blockchain technology can be integrated into system architecture, how alerts can be automated using smart contracts, what consensus mechanisms can be used between nodes, and how confidential data can be stored in a decentralized manner aiming to support the confidentiality, integrity and availability of data.

All articles that include blockchain technologies in their solutions show some similarities in their approach [14, 12, 10, 3]. All solutions chose blockchain to enable efficient and secure transfer of medical data, protect patient privacy and manage access to confidential information. A private blockchain was considered useful in this context as the consensus mechanism between nodes can be simplified, allowing the system to consume fewer resources, which is a potential benefit in resource-constrained IoT environments. None of the solutions relied on commercial blockchains (such as Ethereum), but developed their own stripped-down version that is less resource-intensive. Furthermore, none of the solutions stored the data directly on the blockchain, but used a type of cloud storage for the actual data and only stored the data hashes on the blockchain to improve the resource efficiency and response times of the system. This was complemented by the use of fog nodes to bring compute, storage and network services closer to the IoT devices, reduce latency, improve data protection and increase scalability. Although the same technology was used, the exact implementation specifics and different innovative approaches were used, as described below.

The researchers in [14] have developed a blockchain-based system based on Hyperledger Fabric, a permissioned distributed ledger, and combined it with cloud storage to improve robustness and data CIA (confidentiality, integrity, and availability) in IoT RPM systems. The system architecture comprises four layers: Hyperledger Fabric, Hyperledger Composer, Cloud layer and RPM application layer.

Hyperledger Fabric represents an enterprise-focused private blockchain that provides peer certificate issuance services, a distributed immutable ledger, and a specialized smart contract service called Chaincode. To build the business logic, Hyperledger Fabric uses a connection profile to Hyperledger Composer, which provides a domain-specific language for modeling the business network with its associated assets (medical data), participants (patients and doctors) and transactions. The same service also enables the easy creation of a REST (Representational State Transfer) API service to query the model.

The data from the medical IoT devices is first sent to an IoT Fog Gateway in the local network for data aggregation and temporary storage in case of network unavailability. The gateway then forwards the information to the Hyperledger Fabric running in the cloud. The cloud ledger consists of encrypted medical measurements stored in chained blocks. Once the data is received, the ledger is updated and transactions are stored in cloud storage to improve efficiency and scalability. Patients and doctors interact with the system via the RPM web application, which retrieves the data via the REST service.

Placing a private ledger at the fog layer showed lower server response times and steadier throughput than a cloud-only baseline as device counts and packet sizes rise, indicating that edge aggregation and permissioned consensus can keep end-to-end latency acceptable for wearables.

The model in [12] attempts to eliminate security issues by using blockchain technologies tailored to the resource-constrained environment of IoT. It eliminates the Proof of Work (PoW) consensus mechanism and explores effective cryptographic methods that still match the computing power of IoT devices. An overlay P2P (”peer-to-peer”) network based on a distributed architecture consists of devices called nodes grouped into clusters managed by a cluster head. Communication takes place via the cluster heads which use public key encryption to secure the data and a custom made blockchain consisting of data hashes. The original patient data is not stored in the blockchain but in the cloud to improve the efficiency of the system. The data is stored in identical blocks and encrypted with the patient's public key before its hash is sent to the overlay network. A smart contracts environment is also implemented to enable automatic alerts for healthcare providers. The main cryptographic concepts in this paper are the ARX encryption algorithms and ring signature. ARX algorithms (Addition, Rotation and XOR) are, as the name suggests, a branch of symmetric encryption algorithms that are implemented with simple operations. This makes them suitable for use in computationally less powerful devices. The system uses ARX for double encryption. The medical data is first encrypted with a symmetric key before public key cryptography is used. Another technique, known as ring signature, allows the signer to sign the data anonymously. When the patient wants to sign their data, the system collects other signing requests from other participants and mixes them before sending them across the network. By blending their signature with other signatures in the ring, no one but the signer knows who signed which medical record.

In [10], a privacy-preserving scheme for the secure transfer of patient data and medical diagnoses using a private blockchain and swarm exchange methods is proposed. Swarm is a decentralized P2P file transfer paradigm represented by the nodes in the blockchain network. Files are identified by their hashes and distributed across the network in chunks that are stored in the peer pool. Multiple nodes store the same chunks, ensuring availability, redundancy and resilience to partial outages. The policy, known as Swarm exchange, handles node incentivization that is shared with the nodes which participate in the process by storing the file chunks or providing requested data. In this case, the files the swarm is working with are patient EHRs. Heterogeneous data from the IoT device pool is collected by the aggregator module and converted into an EHR. The data is then encrypted with the healthcare professional's public key. After that, the data itself is stored in the swarm exchange (implemented with InterPlanetary File System (IPFS)), while its hash is sent to the private blockchain, where the transaction is mined and added to the chain. The healthcare professional can then download the file from the swarm exchange after the system confirms through hash checks that the files have not been tampered with, resulting in double-layer security. Exactly the same principles are applied in the reverse scenario, where the doctor makes a diagnosis for the patient.

Swarm availability during uploads remained 10 ms across average EHR sizes, with upload/download times scaling with file size which could be an evidence that off-chain storage with on-chain hashes can bound consensus-induced delays.

Paper [3] presents a private blockchain-based Wireless Body Area Network (WBAN) for the secure and private collection and management of medical data. The main components of the platform are: cloud server, fog device and edge devices. The cloud server represents the medical facility, stores all patient and measurement data and acts as a full blockchain node. The fog devices (usually the patients' smartphones) are also instances of full nodes, which is the opposite of what similar platforms have implemented in the past. The fog devices aggregate communication with the WBAN and are the only components that communicate directly with the cloud service. These devices also manage medical IoT devices and authorize data transfer by performing local checks with the blockchain. By offloading the cloud node, it was possible to achieve process redistribution that offers higher performance, better scalability and more flexibility compared to systems where fog devices do not act as full nodes.

An analysis of the similarities between the approaches examined reveals a number of common features. The blockchain paradigms are used for secure and efficient transfer of medical data as well as robust protection of patient privacy and management of access to sensitive information. The consensus for using private blockchains as opposed to public blockchains stems from the need to minimize resource consumption — a critical consideration in the inherently resource-constrained IoT ecosystems - while allowing an overseeing party to manage system participants. Furthermore, the strategic decision to forgo direct data storage on the blockchain and instead opt for cloud storage with blockchain-based hash references underscores a universal commitment to improving system responsiveness and resource efficiency. Despite these similarities, there are differences in implementation specifics and innovative approaches. Unique cryptographic methods [12], the use of fog computing nodes [14, 3], and novel data management strategies [10] are examples of solutions tailored to specific system requirements and constraints.

While we recognize that network and consensus latencies could still be optimized, the experimental evidence suggests blockchain-based RPM is feasible for low-power IoMT when:

- Permissioned chains avoid PoW,
- Lightweight cryptography (e.g., ARX)ring signatures are used judiciously,
- Data are kept off-chain with fog-level consensus.

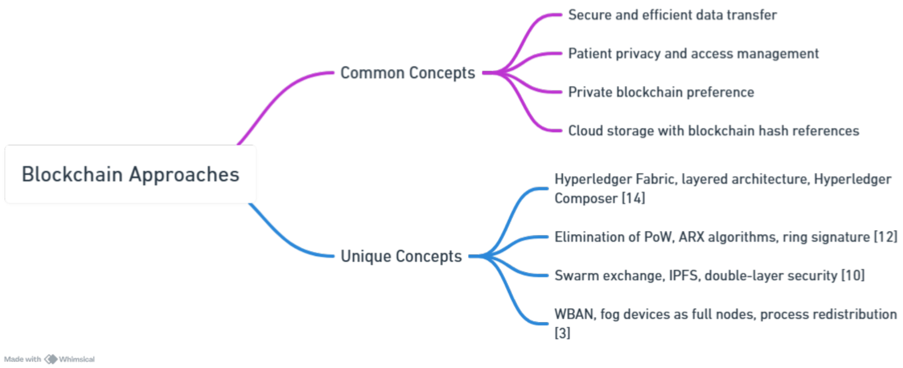
Blockchain Approaches
Common Concepts
Secure and efficient data transfer
Patient privacy and access management
Private blockchain preference
Cloud storage with blockchain hash references
Unique Concepts
Hyperledger Fabric, layered architecture, Hyperledger Composer [14]
Elimination of PoW, ARX algorithms, ring signature [12]
Swarm exchange, IPFS, double-layer security [10]
WBAN, fog devices as full nodes, process redistribution [3]
Made with Whimsical

Fig. 4. Mind map for Blockchain architectures comparison

8. Conclusion

In conclusion, the review of architectural approaches in IoT-based Remote Patient Monitoring (RPM) system design has described diverse approaches aimed at enhancing healthcare delivery through technology.

Four main paradigms—cloud-based, fog-based, hybrid, and blockchain-enabled were identified and analyzed, reflecting ongoing efforts to balance computational efficiency, data privacy, and real-time responsiveness.

Fog-based solutions focus on low latency, local data filtering, quick system response and good patient privacy. For resource-intensive analytics and robust data storage needs, cloud-based infrastructure may be well suited. Hybrid approaches seek to combine the benefits from cloud- and edge-based solutions which enable better scalability while still keeping the overall system costs down by utilizing patients' existing devices. Lastly, the innovative blockchain approaches build on the ability to enable secure data exchange and robust patient privacy, utilizing private blockchains.

Discussed results come from small prototypes, with limited, non-uniform reporting of end-to-end quality of service (Qos) (latency, availability), privacy guarantees, and real-world fault tolerance, making comparisons difficult. Going forward, we see value in standardizing evaluation (benchmarks, shared datasets) and systematically testing live deployments, so missing elements in prototypes' deployment assessments (e.g. interoperability, sustained QoS, energy use, and compliance) are explicitly identified.

The review classifies different approaches for an RPM system design to compare the similarities and differences of the examined systems. This may help researchers and system designers assess which solutions are appropriate for specific contexts and therefore drive innovation of IoT-based healthcare solutions.

Table 1. Compact comparison of IoT RPM architectures and how each addresses core metrics

Architecture	**Latency**	**Scalability**	**Implementation Cost**	**Performance / Throughput**
Fog-based	*Short round-trip time* ([5, 6])	*Offload central server*	*Reusing existing hardware* ([9])	*Processing proximity, server offload*
Cloud-based	*Real-time-capable; load-sensitive* ([13])	*Elastic cloud infrastructure*	*Fewer devices - lower cost*	*Infinitely scalable processing capacity*
Hybrid	*Real-time via ML* ([2])	*Microservices scalability* ([2])	*Reusing existing hardware* ([4])	*High ingress capability* ([2])
Blockchain-enabled	*Fog nodes; off-chain hashes* ([14, 10, 12, 3])	*Fog full nodes* ([3])	*Private chain utilization* ([12])	*Processing redistribution* ([3])

References

1. Arthi, K., Chidhambararajan, B., Revathi, A.R.: A deep investigation of architectural elements and computing technologies for internet of medical things. In: 2022 6th International Conference on Electronics, Communication and Aerospace Technology. pp. 556–563 (2022)
2. Badr, A., Elgazzar, K.: A framework for real-time remote ecg monitoring and diagnoses. In: 2022 5th International Conference on Communications, Signal Processing, and their Applications (ICCSPA). pp. 1–10 (2022)
3. Baucas, M.J., Spachos, P., Gregori, S.: Private blockchain-based wireless body area network platform for wearable internet of thing devices in healthcare. In: ICC 2023 - IEEE International Conference on Communications. pp. 6181–6186 (2023)
4. Chauhan, A., Farmah, K., Goel, A., Gandotra, A.: A novel patient monitoring system using photoplethysmography and iot in the age of covid-19. In: 2021 5th International Conference on Computing Methodologies and Communication (ICCMC). pp. 427–437 (2021)
5. Cheikhrouhou, O., Mahmud, R., Zouari, R., Ibrahim, M., Zaguia, A., Gia, T.N.: One-dimensional cnn approach for ecg arrhythmia analysis in fog-cloud environments. IEEE Access 9, 103513–103523 (2021)
6. Jayson Baucas, M., Spachos, P.: Fog and iot-based remote patient monitoring architecture using speech recognition. In: 2020 IEEE Symposium on Computers and Communications (ISCC). pp. 1–6 (2020)
7. Mahmmod, B.M., Naser, M.A., Al-Sudani, A.H.S., Alsabah, M., Mohammed, H.J., Alaskar, H., Almarshad, F., Hussain, A., Abdulhussain, S.H.: Patient monitoring system based on internet of things: A review and related challenges with open research issues. IEEE Access 12, 132444–132479 (2024)
8. Palli, G.H., Mirza, G.F., Chowdhry, B.S.: Novel iot-based e-health system: Hospital management, telemedicine and quarantine management for covid-19. In: 2022 Third International Conference on Latest trends in Electrical Engineering and Computing Technologies (INTELLECT). pp. 1–9 (2022)
9. Pathinarupothi, R.K., Sathyapalan, D.T., Moni, M., Menon, K.A.U., Ramesh, M.V.: Rewoc: Remote early warning of out-of-icu crashes in covid care areas using iot device. In: 2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). pp. 2010–2013 (2021)
10. Ray, P.P., Chowhan, B., Kumar, N., Almogren, A.: Biothr: Electronic health record servicing scheme in iot-blockchain ecosystem. IEEE Internet of Things Journal 8(13), 10857–10872 (2021)
11. S, D.R., B, A.: Iot revolutionizing healthcare: A survey of smart healthcare system architectures. In: 2023 International Conference on Research Methodologies in Knowledge Management, Artificial Intelligence and Telecommunication Engineering (RMKMATE). pp. 1–5 (2023)
12. Srivastava, G., Crichigno, J., Dhar, S.: A light and secure healthcare blockchain for iot medical devices. In: 2019 IEEE Canadian Conference of Electrical and Computer Engineering (CCECE). pp. 1–5 (2019)
13. Sultana, S., Rahman, S., Rahman, M.A., Chakraborty, N.R., Hasan, T.: An iot based integrated health monitoring system. In: 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA). pp. 549–554 (2021)
14. Zaabar, B., Cheikhrouhou, O., Ammi, M., Awad, A.I., Abid, M.: Secure and privacy-aware blockchain-based remote patient monitoring system for internet of healthcare things. In: 2021 17th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob). pp. 200–205 (2021)

Lan Sovinc (m), MSc, is a full-stack software engineer and IoT developer who completed his master’s studies at the Faculty of Computer and Information Science, University of

Ljubljana. His professional work spans web and IoT system development. He is particularly interested in the design of scalable architectures that combine cloud, fog, and edge computing, as well as the use of AI-driven services. Since 2023, he has been working as a freelancer and participating in projects involving web3 technologies, decentralized finance and artificial intelligence.

Bajec Marko (m), PhD, is a Professor of computer science at the Faculty of Computer & Information Science, University of Ljubljana. He is the Head of the Laboratory for Data Technologies and Iot Demo Center. From 2013 to 2017, he acted as a Vice Dean for Economic Affairs at the Faculty for Computer and Information Science. His research interests related to data technologies include speech and language technologies, information retrieval, web search and extraction, data integration, data management, and data analysis. In the last 20 years, he has led or coordinated more than 40 R&D projects in total amount more than 50 FTE. For his work, he has received several awards and recognitions, such as an Award for contribution in transferring knowledge to industry, Slovenian Society Informatika (2010), an Award for ongoing achievements, international conference on Information Society (2013), a Gold Plaque for outstanding achievements in research and pedagogical work, University of Ljubljana (2014), a Mentor of the Year award, Slovenian association of doctoral students (2014). In his career, he has served as a program board member, program chair or general chair of many international conferences.

Received: February 08, 2025; Accepted: November 20, 2025.