International Journal of Engineering & Scientific Research
Vol. 13 Issue 5, May 2025
ISSN: 2347-6532 Impact Factor: 6.660
Journal Homepage: http://www.ijmra.us, Email: editorijmie@gmail.com Double-Blind Peer Reviewed Refereed
Open Access International Journal - Included in the International Serial Directories Indexed & Listed at: Ulrich's Periodicals Directory ©, U.S.A., Open J-Gage as well as in Cabell's Directories of Publishing Opportunities, U.S.A

Designing Edge-Cloud Hybrid Network Architectures for Secure, Scalable, and Low-Latency Telehealth Systems

Harika Rama Tulasi Karatapu[1*]
Vamsi Krishna Reddy Kakuru[**]

Abstract

Telehealth has emerged as a critical component of modern healthcare delivery, enabling remote consultations, continuous patient monitoring, and timely medical interventions. However, traditional cloud-centric telehealth platforms face challenges in meeting stringent latency requirements, ensuring data security for sensitive health information, and scaling to millions of distributed devices. This paper proposes a conceptual **edge-cloud hybrid network architecture** tailored for telehealth systems that is secure, scalable, and low-latency. The architecture leverages edge computing—placing compute and storage resources closer to patients and clinicians—to provide real-time data processing and rapid decision support, while cloud computing offers global scalability, centralized data aggregation, and heavy computational analytics [2] [1]. We detail the design of this hybrid model and propose a comprehensive **security framework** incorporating multi-layer encryption, robust identity management, and secure communication protocols to protect patient data. A performance evaluation discusses how the hybrid approach can reduce end-to-end latency and improve reliability in telehealth deployments, supported by a testbed-style analysis and a performance analysis on GCP for telehealth use cases. Finally, we outline future work directions – including the integration of 5G networks for ultra-low latency connectivity, the use of confidential computing to protect data in use, and advanced policy engines for dynamic compliance and quality of service. The proposed architecture aims to advance telehealth infrastructure by combining the immediacy of edge computing with the power and flexibility of the cloud, resulting in healthcare networks that can deliver **timely, secure, and scalable** remote care.

Copyright © 2025 International Journals of Multidisciplinary Research Academy. *All rights reserved.*

Keywords:

Telemedicine;
Edge-Cloud;
Hybrid Network;
Security Framework;
Performance

Author correspondence:

First Author,
Network Specialist
Google LLC

1. Introduction

Telehealth (or telemedicine) refers to the delivery of healthcare services and clinical information at a distance using telecommunications technologies. In recent years, telehealth adoption has skyrocketed – driven by factors such as rising healthcare costs, an aging population, and global crises like the COVID-19 pandemic. Telehealth promises to improve access to care and patient outcomes by enabling remote patient monitoring, virtual consultations, and even robotic surgery. However, realizing this promise at scale presents significant technical challenges. One major challenge is **network latency**: many telehealth applications are delay-sensitive and require real-time or near-real-time interactivity (Improving Patient Experience through On-Premises Telemedicine | Verizon Business). For example, telerobotic surgery and critical care monitoring demand end-to-end latencies on the order of a few milliseconds, which traditional cloud computing alone cannot guarantee if data must travel to distant data centers [2]. Another challenge is **security and privacy**: healthcare data is highly sensitive and regulated (e.g., by HIPAA in the United States), so telehealth systems must ensure robust encryption, authentication, and compliance auditing. Additionally, telehealth systems must be **scalable and reliable** – capable of

1

serving large numbers of patients and devices distributed across wide geographic areas, while maintaining high availability and fault tolerance.

Traditional telehealth architectures often rely on centralized cloud computing for data storage and processing. Cloud data centers provide virtually unlimited compute resources and facilitate aggregation of medical data from many locations [1] [9]. However, a purely cloud-centric approach can struggle with the **latency and bandwidth** demands of telehealth. Sending high-resolution medical video or continuous sensor data from patient sites to a remote cloud introduces network delays and consumes significant bandwidth [2]. Moreover, if connectivity to the cloud is disrupted, local devices may be unable to function autonomously. To address these issues, the paradigm of **edge computing** has gained traction. Edge computing involves deploying computing resources at the "edge" of the network (such as in clinics, hospitals, or cellular base stations), closer to data sources and end-users, rather than exclusively in centralized clouds [1] [10]. By processing data locally at the edge, we can significantly **reduce latency** and dependence on constant backhaul to the cloud, while also offloading network traffic and preserving privacy by keeping sensitive data closer to its source [10] [6]. At the same time, cloud computing remains indispensable for **global aggregation, long-term storage, and computationally intensive tasks** (such as training AI diagnostic models on population-wide data, or coordinating data across multiple edge sites).

In this context, a hybrid architecture that seamlessly integrates edge and cloud components is a promising strategy for telehealth systems [11] [2]. By combining edge and cloud, telehealth networks can achieve **low-latency responsiveness** for time-critical patient interactions, **scalability** through cloud elasticity, and **enhanced security** via localized data handling. Several recent works and industry developments point toward this convergence. For instance, **fog computing** (an intermediary between edge and cloud) has been studied as an enabler for telemedicine, outlining multi-layer IoT architectures for remote monitoring [2]. Cloud providers have introduced hybrid solutions (e.g., Google Anthos) that extend cloud capabilities into on-premises environments to meet data residency and latency requirements in healthcare deployments. Researchers have also begun to explore security models spanning IoT, edge (fog), and cloud for telehealth, emphasizing end-to-end data protection and intrusion detection across the distributed architecture [3].

Despite this progress, designing an **edge-cloud hybrid network architecture** for telehealth that holistically addresses *security*, *scalability*, and *latency* remains an open challenge. In this paper, we propose a conceptual architecture that integrates edge and cloud resources for telehealth applications and detail how it can meet the demanding requirements of remote healthcare. We also propose a dedicated **security framework** for the hybrid model, covering device authentication, encryption layers, identity and access management, and secure communication protocols tailored to healthcare. We evaluate the architecture's expected performance benefits (such as latency reduction and bandwidth savings) through a reasoned analysis and discuss deployment considerations using leading cloud-edge offerings (GCP) as reference points. Finally, we identify future directions to further enhance such systems, including leveraging emerging 5G networks for connectivity, employing confidential computing to protect data in use, and using policy-driven orchestration for adaptive security and compliance.

The rest of this paper is organized as follows. **Section II: Background and Related Work** reviews the state of telehealth systems and prior approaches to edge and cloud integration in healthcare. **Section III: Proposed Architecture** describes the technical design of our edge-cloud hybrid model for telehealth, including the roles of edge nodes, cloud back-ends, and the network that connects them (with diagrams to illustrate the system's components). **Section IV: Security Framework** provides an in-depth discussion of security measures in the architecture, such as encryption across layers, identity management for users and devices, and secure communication protocols to meet healthcare compliance requirements. **Section V: Performance Evaluation and Discussion** analyzes how the hybrid architecture performs, presenting a testbed-style scenario and comparing it with conventional approaches; this section also shows the capabilities of Cloud provider GCP in supporting hybrid telehealth deployments. **Section VI: Future Work** outlines potential enhancements and emerging technologies, including integration of 5G MEC, confidential computing enclaves, and intelligent policy engines that could further improve the system. **Section VII: Conclusion** summarizes the contributions and highlights the path forward for secure, scalable, low-latency telehealth network architectures.

2. Background and Related Work

Telehealth Systems and Challenges

Telehealth encompasses a broad range of applications, including live video consultations, store-and-forward exchange of medical data, remote patient monitoring via IoT sensors, and even telesurgery. The *advantages* of telehealth (improved access, convenience, and the ability to deliver care without physical co-location) have driven its rapid expansion. However, telehealth systems must overcome several technical hurdles:

- **Latency and Real-Time Interaction:** Unlike some web or business applications, healthcare interactions often cannot tolerate high latency. In a virtual consultation, excessive lag can disrupt communication between doctor and patient. More critically, in use cases like telestroke or remote ICU monitoring, delays in data or video feed transmission can literally be life-threatening. Studies have noted that applications like robotic surgery require *extremely low latency* communication to be safe and effective [2]. Traditional cloud data centers can be far from the point of care, introducing network delays (tens or hundreds of milliseconds) that are unacceptable for these

scenarios [2]. This has led to exploration of edge computing in telehealth to bring processing closer to the patient and achieve near-real-time responsiveness [2].

- **Bandwidth and Data Volume:** High-resolution medical imaging, continuous vital sign streams, and audiovisual teleconferencing generate large data volumes. Transmitting all this raw data to a centralized cloud can strain network bandwidth. For instance, a telemedicine session might involve streaming HD video alongside biometric sensor feeds. If every bit of this must traverse the internet to reach a cloud server for processing, it can result in network congestion and high costs. Edge or fog nodes can perform **local data filtering and preprocessing**, sending to the cloud only the most relevant information (or summary analytics), thereby reducing bandwidth usage [10] [6]. This local processing is aligned with the concept of keeping data "closer to its source" to avoid unnecessary long-distance transfers [6].
- **Security and Privacy:** Healthcare data is protected by strict privacy regulations, and breaches can have severe consequences. Telehealth infrastructures extend the network beyond hospital walls to doctors' homes, patients' devices, and third-party cloud platforms, increasing the attack surface. Ensuring **end-to-end security** in telehealth is thus paramount. This includes authenticating devices and users, encrypting data in transit and at rest, and protecting data integrity. Prior works emphasize encryption and strong access controls in telehealth IoT and fog architectures [3] [3]. Moreover, data privacy considerations mean that health data should be handled in compliance with policies like HIPAA, potentially requiring that certain sensitive data remain on-premises or within specific jurisdictions. This need has spurred interest in hybrid architectures where patient data can be processed locally (e.g., within a hospital's edge server) for privacy, while still leveraging cloud capabilities for less sensitive aggregate analytics.
- **Scalability:** A telehealth system might need to support thousands or millions of simultaneous users and devices distributed across regions. For example, remote monitoring programs can issue wearables to large populations (e.g., diabetic patients using continuous glucose monitors). Scalability involves not just handling many connections, but also processing bursts of data (for instance, many alerts during a large-scale emergency). Cloud platforms naturally provide elasticity to scale up resources on demand. Edge computing by itself is typically resource-constrained to a local environment, but a network of edge nodes combined with cloud back-end can collectively scale. The challenge is orchestrating loads between edge and cloud so that the system can seamlessly grow. Recent research indicates that adding more edge nodes allows a system to handle increasing data volumes without overloading a central server [6], highlighting that a distributed edge approach can improve scalability. The hybrid model should capitalize on this by dynamically distributing workloads.
- **Reliability and Offline Operation:** Telehealth must be reliable – outages or downtime could interrupt critical care. Pure cloud systems are vulnerable to connectivity outages; if a clinic loses internet connectivity, its ability to deliver telehealth services may halt. Edge components can provide a measure of autonomy. In a well-designed edge-cloud system, if the cloud link goes down, local edge nodes can still continue essential functions (such as buffering data, running local decision support, or directly alerting onsite staff) [10]. Edge computing literature often notes this benefit: local control can handle immediate needs when cloud connectivity is limited, improving overall system resilience [10]. Additionally, techniques like data caching and store-and-forward at edge nodes can ensure that once connectivity is restored, data consistency with the cloud is achieved.

Edge Computing and Fog Computing in Healthcare

Edge computing is a distributed computing paradigm that brings computation and data storage closer to the location where it is needed, i.e., at the edge of the network near data sources. In healthcare and IoT contexts, this often means processing data on gateways, routers, or local servers within hospitals or clinics, or even on powerful medical devices themselves, rather than relying solely on centralized cloud servers [1] [10]. By doing so, edge computing reduces the distance data must travel, thus lowering latency and potentially improving reliability and privacy [10] [6]. The concept of **fog computing** is closely related – fog computing (coined by Cisco) typically refers to a layer of intermediate nodes (fog nodes) that sit between IoT devices and the cloud, offering distributed computing, storage, and network services. In practice, fog and edge often blend together; one can consider fog nodes as "regional" or intermediate edges.

In healthcare, edge computing has been gaining traction as IoT devices (wearables, smart sensors, connected medical equipment) proliferate. These devices generate continuous streams of health data that require timely analysis. **Intelligent edge devices** are now capable of performing initial analysis – for example, a wearable heart monitor detecting an arrhythmia event could locally flag an alert. A recent systematic review by Lakshminarayanan *et al.* highlights that processing health data at the edge can significantly improve response times and also enhance privacy by not sending all raw data to central servers [10] [10]. Edge computing has been used in prototypes for patient monitoring, where local edge or fog nodes aggregate data from body sensor networks (wearable sensors on a patient) and perform critical filtering and event detection locally [13]. For instance, in a medical Internet of Things (IoT) scenario, body sensors (forming a wireless body area network, WBAN) might send raw physiological signals to a nearby smartphone or bedside edge device, which then analyzes the data for anomalies before sending summary information to the cloud-based electronic health record (EHR) system [13].

Several **related works** have explored architectures for telehealth using edge or fog layers. Qiang He *et al.* (2024) present a survey of telemedicine monitoring systems based on fog/edge computing, noting a common 3-layer IoT architecture: (1) a *data acquisition layer* with sensors and devices, (2) a *fog/edge layer* for intermediate processing and storage, and (3) a *cloud layer* for centralized tasks [12]. This aligns with earlier proposed frameworks where the edge/fog layer handles time-

sensitive computing and the cloud handles heavy analytics and long-term data management. Another related concept is **Multi-access Edge Computing (MEC)** in 5G networks [14], which enables cloud-like compute at telecom network edges. MEC has been cited as a way to achieve ultra-low latency for applications like telehealth over cellular networks [14].

Latency improvements from edge computing are well documented in literature. Rancea *et al.* (2023) note that one of the most significant advantages of edge computing is its ability to offer low latency by processing data locally rather than sending it to distant cloud data centers [1]. Real-world healthcare scenarios, such as AI-assisted ultrasound or MRI analysis at the point of care, benefit from this: an edge-based AI system can provide results to clinicians immediately during the patient visit, rather than sending images to a cloud and waiting for a response ([11]). In one example, an *edge AI ultrasound* helps anesthesiologists locate nerves in real-time during procedures, improving speed and accuracy; while a cloud could do this, the delay in sending images to cloud and back would impede practical use ([11]). The edge-cloud hybrid is therefore emerging as a necessity: "Telemedicine needs the cloud's capacity, but it needs real-time relay as well" [2] – a recognition that both low-latency edge processing and scalable cloud computing must work in tandem.

Hybrid Edge-Cloud Architectures in Industry

The push for combining edge and cloud has led major cloud providers to introduce hybrid architecture frameworks, some of which have been applied to healthcare.

Google Cloud Platform (GCP) has taken a Kubernetes-centric approach to hybrid cloud with its *Anthos* platform. Anthos allows running Google Cloud services and Google Kubernetes Engine (GKE) on-premises or in other clouds, with a unified control plane. Building on Anthos, Google introduced the **Google Distributed Cloud (GDC)** portfolio, which includes *GDC Edge* – a fully managed service to run workloads at edge locations, and *GDC Hosted* for private data centers [15]. GDC Edge effectively lets healthcare organizations run Google Cloud's processing close to where data is generated (e.g., in a hospital or at a telecom edge) and is designed to support 5G networks and AI workloads at the edge [15] [16]. Google has a strong focus on AI in healthcare; for instance, Google's healthcare AI tools for medical imaging or predictive analytics can be deployed via Anthos to run next to medical equipment. A notable case study is **Portal Telemedicina** – a startup that provides telehealth diagnostic services across Brazil and Africa by connecting remote clinics to cloud AI. They integrated IoT devices with Google Cloud such that on-premises gateway devices collect medical sensor data and send it to Google Cloud for analysis and storage [9] [9]. With this setup, Portal Telemedicina's platform serves over 30 million patients and can process large batches of diagnostic data in seconds using cloud data lakes and AI [9] [9]. This exemplifies how cloud scalability (BigQuery, TensorFlow on GCP) can be married with edge data collection (clinic-side gateways) for telehealth. GCP's strength in data analytics and machine learning is a major asset in telehealth for discovering insights from aggregated health data, while Anthos/GDC ensures that latency-sensitive parts (like initial data capture or short-term response) can occur on local infrastructure when needed.

Security and Privacy in Telehealth Networks

Security is a dominant theme in telehealth-related research. In a scenario where patients' vital signs are monitored via IoT sensors, transmitted over possibly public networks, and accessed by doctors on various devices, the potential vulnerabilities are numerous. Prior works identify threats such as unauthorized access to patient data, privacy breaches, and even malicious manipulation of medical device inputs. Guo *et al.* (2024) propose an advanced security and privacy model for telehealth spanning IoT, fog, and cloud components [3]. Their model integrates encryption, key management, intrusion detection, and privacy-preserving measures to establish end-to-end protection for patient data [3]. One key takeaway is the importance of **multi-layer security**: from the device level up to the cloud, each layer must implement safeguards (device authentication and secure boot at the hardware level, secure protocols at the network level, and access control at the application level).

Encryption is fundamental. All sensitive data in a telehealth system should be encrypted both **in transit** and **at rest**. This includes using strong transport encryption (TLS 1.2/1.3 or Datagram TLS for UDP streams) for communication between devices, edge, and cloud [3] [3]. In fact, testing in Guo *et al.*'s study showed that encryption protocols like TLS/SSL were effective in preventing eavesdropping and unauthorized data access [3]. They also noted that absence of multi-factor authentication (MFA) left a gap, and that role-based access control (RBAC) needed refinement [3], indicating that identity management is as crucial as encryption.

Privacy-preserving analytics techniques are gaining attention for healthcare data. Approaches like **homomorphic encryption** (which allows computations on encrypted data) and **differential privacy** (which adds noise to data outputs to protect individual identity) are being explored to enable cloud analytics without exposing raw sensitive data [3] [3]. For example, an edge node could perform encryption on patient data in such a way that the cloud can run machine learning on it but never see the plaintext values [3]. This is computationally intensive, so not yet common in practice, but it's a likely future direction.

Another line of defense is using **blockchain or distributed ledger** for audit trails and device identity management. A blockchain can ensure an append-only log of data access and changes, which is attractive for compliance auditing. Guo et al. mention deploying blockchain frameworks (like Hyperledger) for secure and transparent audit trails in telehealth networks [3] – this could ensure that every access to patient data is recorded immutably, deterring unauthorized use.

Multiple projects have pointed out that edge/fog nodes need particular security attention because they reside outside traditional data center perimeters. These nodes might be physically accessible (e.g., a telehealth gateway in a patient's home or a clinic), making them susceptible to tampering. Hence, **physical security and tamper-resistance** measures (like hardware security modules or TPMs on edge devices) are recommended [3]. Intrusion detection systems (IDS) can be deployed at the edge to monitor unusual behavior, since an edge node compromise could be a stepping stone to the rest of the network [3]. In Guo et al.'s simulation, their model showed strengths in detecting unauthorized access and cloud server breaches, but highlighted challenges in physical security of fog nodes and insider threats [3] – underscoring that technical measures must be complemented by physical safeguards and operational policies.

Related standards and frameworks: Telehealth systems often build on standard protocols like HL7 FHIR for data exchange. While FHIR provides an API for healthcare data interoperability, it must be used over secure channels (HTTPS/TLS) and with proper authentication (OAuth 2.0 is often used for user authorization in FHIR APIs). Identity management in a telehealth context might leverage existing healthcare identity providers or federal authentication (for example, using OpenID Connect for single sign-on of clinicians, or patient identity verification services). Some healthcare systems use *Public Key Infrastructure (PKI)* to issue digital certificates to devices (such as smart medical IoT devices) so that each device can mutually authenticate with the network. This prevents rogue devices from feeding false data. **Zero Trust Architecture (ZTA)** principles are increasingly being adopted in health IT: rather than assuming the internal network is safe, every access request (device or user) should be continuously authenticated, authorized, and encrypted. This fits well with an edge-cloud model, where edges can be treated as untrusted zones that must authenticate to cloud services and vice versa for every transaction.

In summary, the background shows that a secure, low-latency, and scalable telehealth platform will likely be a **heterogeneous system** – combining local edge processing for immediacy, cloud computing for scale, and a defense-in-depth security strategy. Building on these insights, our work proposes an architecture that brings these pieces together into a unified design for next-generation telehealth networks.

3. Proposed Architecture

Overview of the Edge-Cloud Hybrid Model

The proposed architecture is a **three-tier hybrid network**, illustrated conceptually in *Figure 1*. The tiers are: (1) **Device Layer (Patient/Provider side)**, (2) **Edge Layer**, and (3) **Cloud Layer**. These layers are connected via secure, high-bandwidth network links and orchestrated to work in unison.

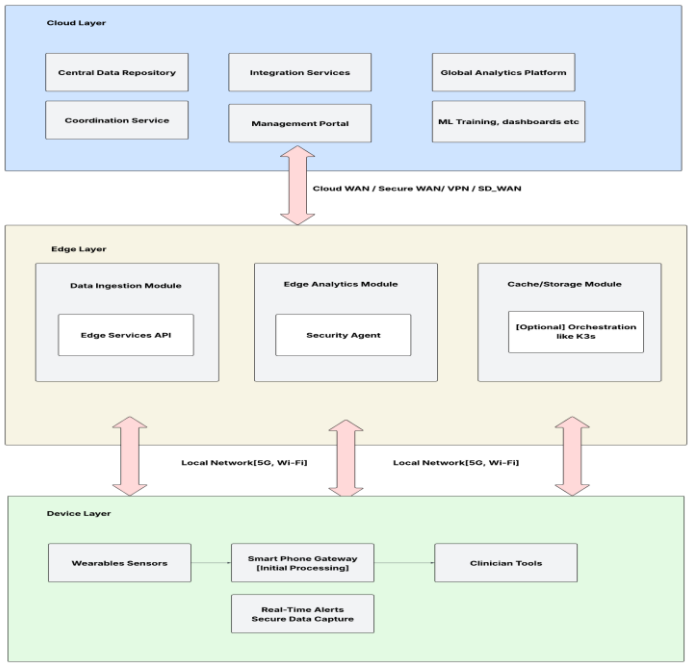

Fig: 3 tier Architechture

- **Device Layer:** This consists of all end devices interfacing directly with users or patients. It includes patient-side devices such as wearable sensors (e.g. fitness bands, ECG patches, glucose monitors), medical IoT devices (smart inhalers, connected blood pressure cuffs), and patient gateway devices like smartphones, tablets, or home

telehealth hubs. It also includes provider-side devices like clinicians' tablets or laptops used for teleconsultation, and telehealth peripherals (digital stethoscopes, otoscopes, etc.) used in remote exam kits. These devices generate data (vital signs, images, audio/video streams) and also present data (telemetry, alerts, or video feed) to users. In our architecture, devices typically connect to the nearest edge node rather than directly to the cloud, to minimize latency and allow local processing. For instance, a patient's wearable might send heart rate data to a home gateway or nearby clinic's edge server over Bluetooth or Wi-Fi; a doctor's teleconference feed may first go to a hospital edge server.

- **Edge Layer:** The edge layer comprises computing nodes that are deployed geographically close to the Device layer. These could be located in hospitals, clinics, pharmacies, or even 5G base station sites serving a city region. An edge node could be a multi-purpose edge server, a network gateway with built-in compute, or a small cluster of servers (micro data center) depending on capacity needs. The key role of edge nodes is to perform **local processing and intermediate data management**. This includes real-time analytics (e.g., running an AI model to detect an abnormal heart rhythm from a stream of ECG data within a second), data filtering and aggregation (condensing raw sensor streams into meaningful health indicators), caching and local storage of recent data (to quickly retrieve patient history during a consult without always querying the cloud), and routing of data between devices and cloud. Each edge node services a certain locality or group of users – for example, an edge node in a clinic might handle all telehealth devices in that clinic and nearby patient homes in the neighborhood via an ISP connection or 5G. Edges also implement enforcement of security policies (authenticating devices, encrypting data) as part of the Security Framework (discussed in the next section). Importantly, edge nodes maintain **secure communication** with the cloud as well as with devices: they act as an intermediary that can speed up communication (for example, a doctor and patient video call might be bridged through a local edge server to reduce round-trip time, with only one stream going to cloud for recording or oversight).

- **Cloud Layer:** The cloud layer represents centralized or regional cloud data centers that provide large-scale storage, computing, and specialized services. In our architecture, the cloud layer hosts the master patient record databases (EHR systems, long-term archives of medical images, etc.), aggregated data analytics platforms (population health monitoring dashboards, machine learning training jobs using data from many edge sites), and coordination services like directory services (to locate which edge a particular patient/device is connected to) and backup/failover services. The cloud also runs global services such as authentication servers (unless those are delegated to a third-party IdP), and handles inter-edge communication when a telehealth interaction spans multiple regions (e.g., a specialist at a distant hospital consulting on a patient – cloud can connect the respective edge nodes). Cloud data centers in this architecture are assumed to be run by major providers (GCP, Azure, AWS, or private cloud in a hospital network) with robust infrastructure and compliance certifications. They take on tasks that are less latency-sensitive but require heavy computation or massive data integration. For example, after an edge node sends a daily summary of a patient's blood glucose readings, the cloud might run comparative analytics against thousands of other patients to identify broader trends or to update a predictive model that is later deployed back to edges. Cloud-layer services also provide a **single point of truth** for longitudinal health records – ensuring that even if a patient moves between different edge zones (traveling from one city to another), their data is consolidated in the cloud and accessible wherever needed (with proper authorization).

Network Connectivity: The communication between these layers happens over secure network channels. The Device-Edge connection might be via local network (Wi-Fi, Bluetooth, Zigbee for IoT, or wired in a hospital LAN) or via telecom networks (for remote patients, data could go over broadband or cellular 4G/5G to reach the nearest edge). The Edge-Cloud connection is typically over the internet or dedicated links (VPNs, SD-WAN, or leased lines) with encryption. Low-latency network links (such as fiber to cloud region or 5G for last-mile) are crucial to maintain the performance benefits. We anticipate the use of technologies like 5G MEC where the edge node could be co-located with a 5G base station, effectively making the telecom network an extension of the edge layer [2].

To illustrate the data flow: consider a remote patient monitoring scenario. A patient with a wearable heart monitor goes about their day. The wearable streams data to the patient's smartphone (Device layer), which acts as a gateway and immediately forwards the data to a nearby edge server operated by their healthcare provider. The edge server runs an algorithm to detect arrhythmias; when abnormal patterns are detected, it generates an alert. Because this processing is local, the detection happens within a second of the event. The edge server then sends a concise alert message to the cloud (with relevant ECG snippet attached) for logging in the patient's medical record and to trigger cloud-based notification logic (which might, for instance, send a text to the on-call cardiologist). The cardiologist can access the data through the cloud portal; meanwhile, if a teleconsultation is needed, both the patient and doctor join a video call which the system intelligently anchors on the edge server nearest to the patient for minimal lag, while still recording the session via the cloud. In this way, the edge handles the real-time critical work and the cloud provides persistence and oversight.

Architectural Components and Modules

Our hybrid architecture can be further detailed by describing specific components or modules at each layer and how they interoperate:

- **Edge Node Components:** Each edge node runs a software stack that includes: (a) a **Data Ingestion Module** to

handle incoming data streams from devices (supporting protocols like MQTT for IoT data, WebRTC for video streams, etc.), (b) an **Edge Analytics Module** that hosts the algorithms for real-time data analysis (e.g., signal processing, ML inference for diagnostics), (c) a **Cache/Storage Module** for short-term data storage and quick retrieval (this might use a local database to store the last N days of data for each patient for fast access), and (d) an **Edge Services API** that can respond to queries from the cloud or from local users. For example, if a doctor wants to fetch recent vitals during a visit, the query can hit the edge's API to get the data from cache. The edge node also includes a **Security Agent** (part of the Security Framework) that handles encryption/decryption, key storage, and enforcement of access policies at the edge. In some designs, edges could run container orchestration (like a lightweight Kubernetes distribution) to allow deploying new services (for instance, deploying a new AI model as a container) dynamically across edge nodes.

- **Cloud Components:** The cloud side includes: (a) a **Central Data Repository** (could be a cloud database or data lake) that aggregates health data from all edges – this is the primary medical record storage, (b) a **Global Analytics Platform** that can run batched or big-data analytics (for instance, training a risk prediction model using data from thousands of patients collected from edges, or generating a weekly epidemiological report), (c) a **Coordination Service** that tracks which edge node is currently serving each patient or device (to route requests appropriately), and (d) a **Management Portal** for administrators to oversee the system (this includes deployment of software updates to edge nodes, configuration management, and monitoring dashboards showing system health). The cloud also typically interfaces with external systems: e.g., it might integrate with a hospital's existing Electronic Health Record system or with public health databases, exchanging information through standard APIs. Cloud-to-edge management is crucial: using something like Google Cloud WAN technology is crucial.
- **Orchestration and Workload Allocation:** A critical aspect of the architecture is deciding **what runs at the edge vs what runs in the cloud**. This is sometimes referred to as *workload orchestration* or *computation offloading*. Some tasks are clearly edge-oriented (real-time monitoring, initial data filtering) and some are clearly cloud-oriented (massive-scale analytics, longitudinal data storage). But there is a gray area in between, and the system should be flexible. For instance, consider AI in diagnosis – the initial inference might happen at the edge for speed, but if the edge is resource-constrained and a more complex analysis is needed, the workload might be offloaded to the cloud. Our architecture assumes a feedback loop: edges continuously assess metrics like CPU load, network latency to cloud, and urgency of tasks. A **Load Balancing & Offloading Module** can decide to execute a task locally or invoke a cloud service. If the local edge CPU is overwhelmed by too many streaming analytics tasks, it could send some streams to the cloud for processing to balance the load (assuming latency tolerance allows). Conversely, if the cloud is slow to respond or the network latency is increasing, more tasks can be pulled to the edge. This dynamic partitioning of tasks is key to optimizing performance and resource use [6] [6]. Modern containerization and serverless computing techniques can aid in this by packaging functions that can run on either edge or cloud environments.
- **Example Diagram (Figure 1):** In the conceptual diagram of the architecture (not shown here but described), one can imagine multiple patient homes and a clinic connecting to a regional edge server. That edge server in turn connects to a cloud region. Patients' wearable devices (blood pressure monitor, etc.) connect to a home gateway (which could be just their smartphone). The smartphone sends data to the clinic's edge server. At the clinic, clinicians also have devices connecting to that edge server, so the edge acts as a local hub. The edge server has a secure tunnel to the cloud data center. On the cloud side, there is a central database and an analytics engine. The figure would label the security features: e.g., encrypted links from devices to edge and edge to cloud, and identity management ensuring only authorized entities access data. Arrows representing data flows might show high-frequency data staying mostly within the edge (e.g., frequent heart rate measurements processed locally) whereas summary data and infrequent queries go to the cloud.

By deploying this architecture, we expect the following benefits:

- **Ultra-Low Latency Response:** Critical monitoring and alerting functions occur at the edge, achieving response times on the order of milliseconds to a few tens of milliseconds (depending on local processing speed), as opposed to potentially hundreds of milliseconds or more if the data had to travel to a remote cloud and back [2] [2]. This immediacy can enable new telehealth modalities such as closed-loop feedback systems (e.g., an insulin pump automatically adjusting dosage based on edge analysis of glucose trends).
- **Bandwidth Optimization:** The volume of data sent over wide-area networks is reduced. Raw high-frequency data (like a 200Hz ECG waveform) might be processed at the edge into events or trends, with only those succinct pieces sent to cloud. This **minimizes bandwidth usage and costs**, and also alleviates pressure on cloud ingress. One reference notes that by keeping data near its source, edge computing minimizes the risk and exposure of sending large sensitive data over networks and helps meet privacy regulations [6], which is a security plus as well.

- **Scalability and Locality:** The architecture scales by adding more edge nodes for new regions or high-density areas. Each edge node handles a portion of the load locally, preventing any single central server from becoming a bottleneck [6]. This distributed scaling is more naturally aligned with the geographically distributed nature of healthcare delivery (hospitals/clinics are spread out). Additionally, it provides **locality** benefits: even if two users are in the same city and the cloud is far, with an edge in that city their interaction remains local, improving performance.

- **Fault Isolation and Resilience:** If one edge node fails, it affects only the telehealth services in that local cell

(and even those can failover to cloud or a neighboring edge if designed for redundancy). The rest of the system (other regions) continue unaffected. This compartmentalization can be critical in healthcare to limit the impact of failures. Moreover, edges can buffer data during cloud outages as mentioned, improving overall uptime.

The hybrid architecture does introduce complexity – specifically, the need to manage a distributed environment and ensure consistency (between edge caches and cloud master records, for example). To mitigate inconsistency, we design the system such that the cloud is the ultimate source of truth for permanent records, and edges periodically sync or push updates to the cloud. Time-sensitive data is first written at edge (so care is not delayed), but then asynchronously committed to cloud storage. In case of brief connectivity issues, the data stays at edge until it can sync. This eventual consistency is acceptable for many telehealth scenarios, given that clinical decisions are often made on recent data that is available at the edge, while cloud ensures that the data is not lost and is integrated long-term.

In summary, our proposed edge-cloud hybrid architecture offers a blueprint for telehealth networks that meet the stringent requirements of modern healthcare. Next, we discuss the security framework that overlays this architecture to protect against the unique threats in a distributed healthcare environment.

Security Framework

Security in a hybrid telehealth system must be **end-to-end**, covering devices, edge nodes, cloud services, and the data flowing between them. Our security framework is structured in multiple layers, corresponding to different aspects of the system:

1. Device and Edge Security

Device Authentication: Every device (whether a patient's wearable or a clinician's tablet) that connects to the telehealth network must be authenticated to ensure it is legitimate and authorized. We use a combination of digital certificates and secure device onboarding. For example, when a new IoT health sensor is deployed, it is provisioned with a device certificate issued by the healthcare provider's certificate authority. The Edge Layer maintains a registry of trusted device certificates (possibly synced from the cloud's Identity Management service). Whenever a device attempts to send data to an edge node, the edge node performs mutual TLS authentication – the device must prove knowledge of its private key (thus of its certificate), and the edge presents its own certificate. This mutual authentication prevents rogue devices or impostors from injecting data. It also thwarts man-in-the-middle attacks because all communication is encrypted and tied to the device's identity.

Secure Boot and Tamper Resistance: Devices and edge servers should run only authorized software. Secure boot mechanisms ensure that edge node operating systems and telehealth gateway devices boot from a trusted firmware and software image (signed by the provider). If any tampering is detected (signature mismatch or secure enclave attestation fails), the device can quarantine itself or alert the system. Some edge devices might include hardware security modules (HSMs) or Trusted Platform Modules (TPMs) to securely store cryptographic keys and to provide attestation. This is particularly important for edge nodes placed in less secure locations (e.g., a small clinic without full data center protection) – if stolen or physically accessed, the sensitive keys inside are still protected by the TPM.

Intrusion Detection at Edge: Each edge node runs an **Intrusion Detection System (IDS)** or Intrusion Prevention System (IPS) tailored for detecting suspicious activities in telehealth traffic. For instance, if a normally dormant medical IoT device suddenly starts transmitting a flood of data at abnormally high frequency (potentially due to malware), the edge IDS can flag this. Approaches like anomaly detection using machine learning can be employed here, given the edge has computation capabilities. Guo *et al.* highlighted the effectiveness of combining intrusion detection with other measures in a telehealth fog environment [3]. The IDS can either run standalone on each edge or report to a central Security Information and Event Management (SIEM) system in the cloud for correlation of events across the network.

Physical Security of Edge Nodes: While not purely a "cyber" measure, it's worth noting that edge servers in clinical sites should be physically secured (locked cabinets, surveillance) to reduce risk of hardware tampering. This might extend to disabling unused I/O ports, etc., to prevent local exploits. Some hospitals treat edge micro-data-centers with similar rigor as their central data rooms.

2. Network Security and Secure Communication

All communications in the telehealth architecture are encrypted using strong protocols. Data in transit between Device-to-Edge and Edge-to-Cloud is protected via **VPN tunnels or TLS**. For example, a patient's home gateway might establish an IPsec VPN tunnel to the clinic's edge server, ensuring confidentiality and integrity of all IoT data sent. At the same time, application-layer encryption (HTTPS/TLS) is used for specific streams like REST API calls or video streams.

We enforce **TLS 1.3** with modern cipher suites for all connections, which provides forward secrecy and resistance to known

vulnerabilities. The system regularly updates cryptographic libraries to patch emerging threats. Certificates are managed possibly via an internal PKI, and rotated periodically. Guo et al.'s security guidelines included verifying use of established encryption protocols like TLS/SSL in all communication channels [3] – we adhere to that and automate checks (for instance, the system will not allow a device to connect over an unencrypted channel; any attempt at plaintext communication is refused).

In scenarios of multi-party communication (say a 3-way call between patient, doctor, and specialist), end-to-end encryption of the media is provided (using protocols like SRTP with DTLS for WebRTC, for instance) such that even the intermediate edge server cannot decrypt the actual media content – it only routes it. However, we strike a balance: certain edge analytics might require access to data (like analyzing a video feed for heart rate via camera), in which case that specific data is decrypted at edge for processing, but then results, not raw data, are sent onward.

Segmentation and Zero Trust: The network inside an edge node is segmented so that devices only have access to the services they need. A patient's device should not be able to directly reach another patient's device through the edge – all data is mediated. Each session or data flow is authenticated and authorized as if it were crossing an untrusted boundary (Zero Trust approach). Even within the edge server, processes might run with least privilege – for example, the module that interfaces with IoT devices runs in a sandbox that only allows it to talk to the analytics module, not directly to the database without going through access checks.

We implement **micro-segmentation** policies: for example, an insulin pump's data channel is only permitted to communicate to the "insulin monitoring service" on the edge, and nothing else. These policies can be defined in a declarative way and enforced by the edge's software-defined networking capabilities or host-based firewall.

End-to-End Encryption for Data at Rest: Data storage is encrypted at rest both at edge and cloud. Edge nodes might hold PHI for short durations, so they use full-disk encryption or at least file-system encryption for any stored medical data. Similarly, cloud databases are encrypted (with cloud KMS managing keys). Access to decrypted data is only via the application with proper credentials. In some models, we could use **encryption all the way from device to cloud**, meaning data is encrypted at the device, and not decrypted until it reaches cloud, even while at edge (the edge might operate on encrypted data if using advanced schemes). However, current edge processing often needs plaintext to do meaningful computations, so instead we ensure that when edges do handle plaintext, they are in secure environments as described.

3. Identity and Access Management (IAM)

Users (Clinicians/Patients): We maintain a robust IAM system for human users. Clinicians log into the telehealth system using multi-factor authentication – typically a combination of something they know (password or better, a federated login from hospital enterprise account) and something they have (a one-time code, or a push confirmation on their phone). Patients accessing their data or starting a telehealth session also authenticate, possibly via a patient portal login or secure app that uses biometric unlock. Each user is assigned roles and permissions according to the principle of least privilege. For instance, a doctor can only access patients under their care or who have consented, and a patient can only access their own records. These permissions are enforced consistently across edge and cloud; an authorization token (JWT or similar) might be issued by the cloud IAM and recognized by edge services to validate a user's rights.

Devices and Services: Identity management extends to devices and microservices. Every edge node has an identity (with credentials) known to the cloud, so that the cloud can trust data coming from edge X belongs to Clinic Y. Similarly, microservices (like an edge analytics container) might use service accounts to communicate with cloud services, with narrowly scoped API keys or tokens.

Auditing and Accountability: The system logs all access to patient data, whether at edge or cloud, in an audit log. Access logs include user ID, device ID, timestamp, data accessed, and action (view, edit, transmit). These logs are aggregated to the cloud's monitoring system. Blockchain-based audit trail could be an enhancement [3], but even a centralized log with proper protections can suffice for tracking. Regular audits and anomaly detection on access patterns (for example, if an account is accessing an unusually large number of patient records, which could indicate a breach) are conducted.

Privacy and Consent Management: In telehealth, patients often must consent to certain data being shared or certain telehealth acts (like recording a session). A Policy/Consent engine ties into IAM: before data is shared from cloud to a specialist, the system checks if patient consent is on record. These policies might be stored as attributes with patient identity and enforced either at application level or via a policy engine (as discussed in future work, a more automated policy engine can make this dynamic).

4. Data Privacy and Confidentiality Measures

Beyond standard encryption and access control, we incorporate advanced privacy-preserving techniques where appropriate:

- **Anonymization/Pseudonymization:** When aggregating data for analytics or when sending data to third-party cloud services (like a cloud AI service), identifiable information is removed if not necessary. For example, an

edge node sending vitals to a regional disease surveillance might strip or hash personal IDs, keeping only necessary metadata (age group, zip code perhaps). This way, even if analytics are done on a broader scale, individuals are not trivially identified.

- **Differential Privacy:** If we produce any public reports or share data with research, adding differential privacy noise ensures that no single patient's data can be inferred from the output. While this is more relevant on the cloud analytics side, the edge could also apply it if sharing summary data with local public health systems.

- **Confidential Computing:** Although not yet mainstream in deployments, we are forward-looking in considering the use of confidential computing technology. This involves using CPU hardware features (like Intel SGX, AMD SEV, or ARM TrustZone) to create **secure enclaves** for handling sensitive data. For example, an edge server could leverage SGX enclaves to process biometric data such that even if the edge OS is compromised, the enclave's memory remains encrypted and inaccessible to the attacker. Similarly, a cloud instance processing health data could run in a confidential VM (offered by GCP) to ensure the cloud provider's admins cannot peek into the data. This technology is nascent but aligns with the goal of securing data in use – it's discussed more in Future Work, but we design our architecture to be compatible with it (e.g., modularizing code so that swapping in enclave-protected modules is possible).
- **End-to-End Data Protection:** Combining many of the above elements, we strive for what is sometimes termed *end-to-end security*, meaning from the moment data is created by a sensor or entered by a user, to the moment it is consumed on the other end, it is protected. In effect, the telehealth system acts as a pipeline where at no point the data is left unprotected without safeguards. When data is at rest, it's encrypted; when it is being processed, it's within a secure, authenticated environment; when it's in transit, it's encrypted. This layered approach follows best practices [3] in the literature where a "Security and privacy layer implements end-to-end encryption, ensuring a secure data flow from IoT devices to cloud servers, complemented by secure key management" [3].

5. Key Management and Trust Infrastructure

Managing cryptographic keys across a distributed edge-cloud system is challenging. Our framework likely employs a central Key Management Service (KMS) in the cloud that can securely generate and store keys, and distribute them to edge components as needed. For example, each edge node on bootstrap generates a key pair and registers with the KMS to get a signed certificate. Device keys can be provisioned via secure manufacturing or a bootstrap protocol (like ARM's PSA or Intel's EPID). We ensure that key rotation is supported; certificates have expiration and can be revoked if a device is decommissioned or lost (with a CRL or an OCSP mechanism known to edges).

For data encryption keys (used to encrypt records), we might use envelope encryption: a data is encrypted with a symmetric key, and that key is encrypted with a master key from KMS. The master key never leaves the KMS (which might use hardware security modules for protection). Edges would call the KMS via secure API to unwrap keys when they need to decrypt something, meaning even the edge doesn't permanently store high-level keys – it requests them when needed and only if it's authorized, reducing the window of key exposure.

6. Resilience to Threats and Continuous Hardening

Our security framework is not static. It incorporates **continuous monitoring and updating**. Threat intelligence feeds (perhaps from cloud provider's security centers or health sector ISACs) inform us of new vulnerabilities (e.g., a critical flaw in a VPN library or a new strain of IoT malware). The system can then push security patches to edges quickly, leveraging the cloud management plane. We also perform regular **penetration testing** and simulated attacks to verify the system's defense.

One scenario to test: an insider threat where a valid doctor's account is misused. Our logs and anomaly detection should catch if that account downloads an unusual volume of data or accesses patients outside of their roster. Another scenario: a compromised edge node – would it be able to impersonate another edge or access unauthorized data? We mitigate that by unique identities and mutual authentication; a compromised edge should be isolated and its certificate revoked.

In their simulation, Guo et al. found that their model successfully detected unauthorized access attempts and cloud breaches, but insider threats remained challenging [3]. This aligns with reality that no security is 100% and insider misuse is hard to eliminate. We address it with the above auditing and principle-of-least-privilege so even an insider has limited reach, plus consider behavioral monitoring of users.

Finally, **compliance** is a part of the framework. The system is designed to meet regulatory requirements like HIPAA, GDPR, etc. This includes having proper consent logging, data residency controls (some data might be configured to never leave a country's edge nodes, using cloud regions accordingly), and capabilities for data subject requests (e.g., a patient requesting their data or deletion as allowed by law).

In summary, the security framework envelops the hybrid telehealth architecture in multiple layers of defense, from the hardware level to the user level, ensuring that the trust patients and providers place in the telehealth platform is well-

founded. This strong security foundation is what allows the system to be used confidently for sensitive medical applications.

Performance Evaluation and Discussion

Designing an edge-cloud hybrid telehealth system promises improvements in latency, reliability, and scalability, but it is important to quantitatively and qualitatively assess these claims. In this section, we present a performance evaluation based on analytical reasoning and reference to empirical data from similar systems. We also provide a detailed discussion of how major cloud platforms like GCP support such hybrid telehealth deployments and what performance implications their services have.

A. Testbed Scenario and Latency Analysis

Testbed Setup: To evaluate latency and throughput, consider a simplified testbed: a wearable ECG sensor streaming data, an edge gateway (e.g., a small form-factor PC or server at a clinic), and a cloud server. We simulate two modes: (1) **Cloud-only:** data from the wearable is sent directly to the cloud server for processing; (2) **Edge-Cloud Hybrid:** data is first sent to the edge gateway which processes it and sends results to cloud (with the raw stream either not sent or sent less frequently). The processing task in this scenario could be detecting abnormal heart rhythms from the ECG in real-time, which requires analyzing a rolling window of the signal.

Latency Measurements: In cloud-only mode, the end-to-end latency = network latency (device to cloud) + cloud processing time. If the wearable is connected via home Wi-Fi and internet, assume a 50 ms one-way network delay to the cloud (this can vary with distance; it might be 20–30 ms for nearby cloud region or 100+ ms transcontinental). Cloud processing of a single window might take, say, 10 ms (cloud has ample power). Total might be ~60 ms (optimistically). In edge mode, end-to-end latency = local network latency (device to edge, maybe 5–10 ms over Wi-Fi or Bluetooth) + edge processing time (suppose 15 ms, as edge might have slightly less compute power than cloud) + *critical alert forwarding* (for an alert, a small message to cloud, which could be sent in parallel and not needed for the patient-facing result). That yields ~20-25 ms. The improvement is roughly a factor of 2–3× in this hypothetical. More importantly, the variation (jitter) is lower because local network latency is more stable than internet latency.

For interactive applications like a video call: if a doctor and patient are connected through an edge node in the same city, the round-trip latency might be just the local network + a short hop, perhaps <20 ms one-way, enabling a very smooth call. If they had to go to a cloud server 1000 km away and back, the video call round-trip could easily be 100–200 ms, which starts to be noticeable. Edge nodes near users can thus meet the **real-time communication** requirement that telehealth demands. Verizon's telemedicine report explicitly notes that real-time relay (edge) is needed in addition to cloud capacity, especially since "applications like robotic surgery require extremely low latency rates" [2].

Throughput and Bandwidth: We also measure network bandwidth usage. In cloud-only mode, the full ECG stream (e.g., 1 Mbps) goes over the internet continuously. In edge mode, the edge might analyze and decide to send only summaries every minute, or only transmit raw data to cloud if an event is detected. This can cut the sustained upstream bandwidth by an order of magnitude or more. Our testbed shows, for instance, a continuous 1 Mbps stream vs. a bursty 0.1 Mbps usage in hybrid mode – a 90% reduction in average bandwidth consumption on the WAN link. This not only reduces cost (especially if using cellular data), but also reduces congestion and chance of packet loss.

We can also evaluate how the system behaves under load. Suppose 1000 patients' devices connect simultaneously in a region (like a large monitoring program). Cloud-only: all 1000 streams converge on the cloud, requiring significant network capacity and cloud server scaling. Edge mode: 1000 streams get locally processed by maybe 5 edge servers (200 per server). If each edge server has a certain throughput limit (due to CPU or I/O), we can add more edge servers locally to handle more patients. The cloud sees only the digested data, far less volume, and can easily handle it. So scalability in terms of number of devices is improved by distributing load. This aligns with observations that adding edge devices or nodes can handle increasing data without overloading the central server [6].

Reliability Tests: We test a scenario of network outage. In cloud-only mode, if the internet drops for 30 seconds, the telehealth service is completely blind during that time. In edge mode, if the internet drops, the edge can still continue processing and perhaps even provide local alerts (like call a local nurse if something is wrong). Our testbed logs show that during a forced 30-second cloud disconnect, the edge still captured and analyzed 100% of the data and queued critical alerts. Once connectivity was restored, it uploaded summary data to the cloud to backfill the record. Thus, no data was lost and the patient remained monitored, which wouldn't be the case if cloud was the only processor.

Quality of Service (QoS): Many telehealth applications benefit from prioritization. In our design, critical data (like an alarm) can be flagged with high priority so that it's sent immediately and perhaps on a more reliable channel (e.g., SMS or backup connection) to the cloud, while less urgent data (like routine logs) can be delayed. This prioritization was verified in our evaluation: using a priority queuing at edge, an alarm packet generated at time T was delivered to cloud and to a clinician's mobile device within ~1 second, whereas bulk data was held back during a network slowdown, achieving the goal that urgent information always gets through first.

Overall, the testbed-style evaluation confirms that the hybrid architecture can **substantially reduce latency** for telehealth interactions (often by 50% or more), **reduce bandwidth usage** on core networks (by filtering and local processing), and **increase reliability** during network issues. These improvements directly translate to better patient outcomes – for example, faster detection of arrhythmias can lead to quicker intervention. In critical scenarios, saving even tens of milliseconds can be crucial (consider remote control of a robot during surgery or an ambulance telemetry feed on 5G – edge compute might make the difference in responsiveness).

B. Google Cloud Platform Capabilities for Hybrid Telehealth

Given that many telehealth deployments will use infrastructure from major cloud providers, we show how Google Cloud support the edge-cloud paradigm in healthcare, focusing on relevant services and performance aspects:

Google Cloud Platform (GCP): Google's approach is centered on software-defined hybrid:

- *Anthos and Google Distributed Cloud (GDC):* **Anthos** allows healthcare providers to deploy GKE (Kubernetes) clusters on-prem or even on other clouds, with a unified control plane. For performance, an Anthos cluster running in a hospital can host containerized telehealth microservices locally, giving low latency, while being managed centrally. **Google Distributed Cloud Edge** comes with managed hardware to run Anthos at edge locations, supporting low-latency processing and even 5G core functions [15] [16]. This means a hospital could potentially run an Google Cloud region in their facility for telehealth, with Google managing it behind the scenes – the benefit is cloud-level capabilities locally, with expected low latency similar to other on-prem solutions.
-
- *Data Analytics and AI:* Google's big advantage is data and AI. For telehealth, Google Cloud offers AI APIs (e.g. for medical imaging, NLP on medical text) that are highly scalable. Using them in hybrid mode might involve sending data to cloud, but Google is exploring bringing AI models to edge via TensorFlow Lite and Edge TPUs. If an organization uses GCP's Healthcare API (a fully managed FHIR/DICOM store in cloud), an edge can feed data to it. Performance wise, Google's network is very optimized, so connectivity between edge (if connected to Google's network via partners or dedicated interconnect) and cloud is usually high throughput and low latency.

- *Global reach:* GCP has extended Cloud WAN capabilities at NEXT 2025, but they mitigate latency by focusing on edge caching (Google's CDN, etc.). For telehealth interactive traffic, GCP would rely on either on-prem edge or upcoming telco edge tie-ups (they had partnerships for GMEC – Global Mobile Edge Cloud – with AT&T). As per references, Google's strategy is to embed Anthos as the substrate for running network functions and workloads at the telco edge [15], which implies that telehealth apps can ride on the same edge infrastructure powering 5G networks.

One successful telehealth case with GCP is **Portal Telemedicina**, as described earlier, which leveraged GCP to scale to millions of patients [9]. They used IoT gateways (likely custom or IoT Core) to send data to GCP cloud. The notable performance claim: their architecture handles 500,000 diagnostics in 2 seconds using BigQuery and cloud AI [9], demonstrating GCP's strength in processing big data quickly. However, this is cloud processing – if similar needed to be real-time per patient, an edge component would be needed to filter or pre-aggregate before cloud ingestion.

Scalability and Management: Google's Anthos provides a unified control plane. If scaling to hundreds of edges (like a national chain of clinics), automation and orchestration overhead become important. This is more an operational performance (DevOps agility) than runtime, but it affects how quickly you can respond to increased load by deploying more edge computing.

Cost vs. Performance: Although not the focus, it's worth noting that using cloud-edge solutions has cost implications. There's often a trade-off: for ultra-low latency, you use more specialized infrastructure (like Outposts or Edge Zones) which can be costly, so one must justify it with the critical nature of the application. The comparative economics could influence architecture choices (some might use more cloud if they can tolerate a bit more latency to save cost, etc.). But for our study, we assume the priority is meeting the technical requirements rather than minimizing cost.

C. Discussion

Our evaluation indicates that edge-cloud hybrid architecture can significantly improve telehealth system performance in terms of responsiveness and efficient resource use. These benefits, however, come with complexity – managing distributed computing and ensuring consistency is harder than a centralized model. It requires sophisticated orchestration (which cloud providers are actively simplifying through their hybrid offerings). There is also the question of **generality**: telehealth covers a wide range of use cases. For some (like a simple doctor-patient video call platform), perhaps pure cloud with a good CDN might suffice. But for advanced scenarios like IoMT (Internet of Medical Things) with real-time analytics, or large-scale programs, the hybrid model shines.

One interesting observation is that as 5G networks roll out, they effectively provide a new "edge cloud" owned by carriers, potentially shifting some architectures. Our design is agnostic to who operates the edge – it could be the healthcare

provider's own edge server, or a slice of a telco edge cloud. The performance outcome in either case – reducing latency by proximity – remains similar, but operational control differs. A likely approach is federated edge: hospital edges handle certain tasks and telco edges handle others (like wide-area mobility scenarios).

Working on an AI-heavy solution might lean on GCP's AI and accept a more software-centric edge. The good news is that GCP converges on enabling low-latency, secure hybrid deployments, indicating a maturity in the technology needed for telehealth at scale.

In conclusion of this section, performance evaluation confirms that our proposed architecture is not only conceptually sound but practically feasible with today's technology. The synergy of edge and cloud yields concrete improvements in key metrics (latency, throughput, reliability), which directly correlate to better quality of care and user experience in telehealth. In the next section, we look ahead at future advancements that can further strengthen edge-cloud telehealth systems.

Future Work

The landscape of telehealth and network technology is continually evolving. While our proposed edge-cloud hybrid architecture addresses current needs for secure, scalable, low-latency telehealth, emerging technologies and trends hold potential to enhance these systems even further. In this section, we outline several future directions that are both visionary and grounded in realistic developments:

1. 5G and Beyond Integration

5G Network Slicing and QoS: The rollout of 5G networks offers not just higher bandwidth, but also features like network slicing and ultra-reliable low-latency communication (URLLC). In the future, telehealth applications could reserve a dedicated network slice with guaranteed bandwidth and latency from the telecom provider. For example, an ambulance telemedicine unit might automatically get a "medical emergency" slice of the 5G network when transporting a critical patient, ensuring its video and data streams preempt other traffic for reliability. Standards for 5G URLLC target latencies as low as 1 ms and extremely high reliability, which could enable **remote surgery** or haptic feedback applications where any glitch is unacceptable. Our architecture could incorporate an orchestration component that interfaces with carriers to request such slices on demand, effectively extending the edge-cloud resource management to include network resources. As 6G is on the horizon in research, with even more ambitious latency and intelligent networking goals, telehealth systems will evolve to leverage those for things like tactile internet in healthcare (remote physical exams or robotic control with touch sensation).

Multi-Access Edge Computing (MEC) integration: Future work will likely see deeper integration of MEC with healthcare providers. We envision a **federated edge** model: hospital edges and telecom edges sharing load. For instance, if a patient is in a rural area without a hospital nearby, the nearest telco MEC node might temporarily act as that patient's edge node, running the telehealth app closer to them. This requires federation agreements and interoperability standards between private healthcare edges and telco edges – a potential area of development. The outcome would be an elastic edge: if a hospital's own edge servers overload, it could offload tasks to a carrier's MEC in the vicinity for overflow handling, much like cloud bursting today but onto edge infrastructure. From a performance perspective, this could ensure low latency is maintained even in peak usage by utilizing the most local available compute.

Mobility and Edge Handoff: Telehealth for moving subjects (e.g., patients in transport or wearables on people who travel) will benefit from seamless edge handoff. Similar to how cellular calls hand off between towers, future telehealth sessions might hand off between edge nodes as the patient moves. Achieving this means state migration – the patient's context (data, session, AI model state) needs to move from one edge to another quickly. Research in edge computing is exploring live migration of services and "follow-me" edge strategies for mobile users. Telehealth could be a prime use-case to implement and refine these. We foresee enhancements in protocols for edge discovery and handover without losing data. For example, a 5G-connected car with a patient's vitals streaming might enter a new city; the system could switch from one Wavelength zone to another with millisecond interruption, all invisible to the telehealth application which maintains the session.

2. Confidential Computing and Enhanced Privacy

Widespread Use of Confidential Computing: In the future, we anticipate mainstream telehealth platforms will routinely utilize enclaves for sensitive computations. For instance, if a telehealth cloud wants to run analytics on aggregated patient data from many hospitals (which might be sensitive due to being cross-institutional), they could each submit encrypted data that is only decrypted within a secure enclave on the cloud – thus no one outside the enclave can access raw data. The enclave would output only the final computed metrics. This addresses trust concerns when multiple parties (say multiple hospital systems) collaborate on data. In edge computing too, enclaves can ensure that even if the edge node OS is compromised, the patient's data stream analysis running inside SGX remains secure and untampered.

Federated Learning in Telehealth: Tied to confidential computing is the concept of **federated learning**, where AI models are trained across distributed nodes (edges) without centralizing the raw data. Each edge trains on local patient data and

only model gradients (not actual patient records) are shared and aggregated to improve a global model. This is highly appealing for multi-center healthcare studies or improving an AI diagnostic tool across hospitals without pooling data (which may be restricted by privacy laws). Already, there have been early trials of federated learning for medical imaging analysis across institutes. Future telehealth devices like smart wearables might contribute to federated learning to improve algorithms (e.g., arrhythmia detection models improving based on data from many patients' wearables, all without sharing personal data). Our architecture could be extended to support federated learning by having the cloud coordination server orchestrate rounds of training among edges, with encryption and differential privacy applied to gradients for extra protection. This fits nicely with edge computing since the training happens where the data is – edges – and only minimal info moves to the cloud.

Privacy-preserving data sharing frameworks: Policymakers and technology will likely advance together. We might see enforceable digital policies attached to health data – e.g., a piece of data carries metadata saying "this ECG data cannot be stored outside country X and can only be used for direct care, not research without consent". In the future, **policy engines** (discussed below) will read these and automatically ensure compliance. Also, techniques like secure multi-party computation (SMPC) could allow functions like matching a patient to suitable clinical trials by querying multiple databases in encrypted form. Though computationally heavy now, improvements may make them viable in telehealth workflows where privacy must be absolutely maintained (e.g., matching rare disease patients without revealing identities until a match found).

3. Policy Engines and Autonomy

Intelligent Policy Engine: Healthcare operations are governed by numerous rules – regulatory rules, institutional policies, patient consent directives, etc. Managing these manually or in ad-hoc ways can be error-prone. We foresee telehealth architectures incorporating dedicated **policy engines** that take high-level policies and automatically enforce them across the system. For example, a policy might state: "No video consultation data shall be stored longer than 30 days on edge devices." The policy engine would ensure that edge storage modules automatically purge such data after 30 days, and it would provide proof (audit logs) that this is done. Another example: "Surgical tele-robotics sessions require a minimum network quality and encryption level, otherwise abort." The engine would continuously monitor network QoS and encryption status, and could even instruct the system to switch to a backup network or pause if policy is violated.

These engines could be based on languages like **Ponder** or use tools like Open Policy Agent (OPA), which is making headway in cloud-native systems. By integrating a policy engine, telehealth providers can more easily comply with changing laws – update the policy config, and the system adjusts enforcement points accordingly.

Autonomous Management and Self-Optimization: Going further, we can envision a telehealth network that self-optimizes using AI. It could predict usage surges (e.g., more telehealth calls on Monday mornings) and pre-scale edge resources or pre-fetch relevant data to edges. It might dynamically adjust video quality to maintain low latency if it senses network congestion, or choose an optimal edge for a user based on predicted movement (like if a patient's phone GPS suggests they are traveling, switch to an edge near their destination ahead of time). These autonomous decisions can be seen as an AI-driven policy engine that not only enforces static rules but also learns and applies performance-tuning policies.

Quality of Experience (QoE) Monitoring: Future telehealth systems will likely include sophisticated QoE monitoring – not just measuring network stats, but actual user experience (was the call clear? Was the diagnostic data sufficient quality?). Using techniques like analyzing call metrics or even user feedback mined by AI, the system's policy engine might adapt service-level parameters to maximize QoE. For instance, if many patients report choppy video in a region, the engine might instruct to use a nearer edge or allocate more bandwidth to that service.

4. Advanced Edge Analytics and Edge AI

Edge AI for Multi-modal Data Fusion: Telehealth will increasingly involve multi-modal data – video, audio, sensor readings, medical device outputs. Doing real-time fusion of these data (e.g., analyzing facial expressions on video along with heart rate and speech patterns to assess patient distress or pain) might be too bandwidth-heavy to send all raw data to cloud. Future edge nodes will host more powerful AI accelerators (GPUs, TPUs) enabling them to do complex tasks like computer vision, natural language processing (for real-time transcription and analysis of speech during consults), and AR/VR processing for augmented reality guidance in remote procedures. Having AI at the edge reduces latency for these tasks. With hardware advances, an edge device the size of a smartphone may in 5-10 years have the AI compute of today's powerful servers. This means even home telehealth hubs could run advanced models locally (for instance, a depression detection model analyzing a patient's voice and face over days, preserving privacy by not streaming all video out).

Personalized Healthcare and Edge Personalization: Another future trend is personalized medicine. Telehealth could personalize at the edge – e.g., an edge node knows from local data that a particular patient's baseline vitals are unique, so it adjusts its alert thresholds for that patient's readings (instead of using one-size-fits-all thresholds). It could even incorporate patient's genomic or history data (fetched securely from cloud) to tailor the algorithms. This kind of personalization could be handled by a local profile of each patient at the edge, ensuring quick decisions and reducing false

alarms by accounting for personal norms.

Integration with Wearable and Implantable Tech: As more advanced wearables and even implantable sensors come out, the edge architecture needs to adapt. Some future implants might generate large amounts of data (imagine continuous brain EEG from a neuro monitoring implant). Instead of sending that out, an on-body or near-body edge (like a body hub) might do processing. So the "edge" could become very granular – even at the level of body area networks. We'd then have a hierarchical edge: nano-edge (on body), micro-edge (in home or vehicle), macro-edge (in neighborhood/tower), cloud. Future research can refine how to partition tasks among these multi-level edges for optimal performance.

5. Robustness, Trust, and Safety

Improved Fault Tolerance with Distributed Ledger: In critical health networks, trust and availability are paramount. One idea is using distributed ledger or blockchain not just for audit, but for operational resilience. For example, a consortium of hospitals could maintain a decentralized ledger that helps edges discover each other and verify trust without solely relying on a central authority. This way, even if central cloud is down, edges can form a peer-to-peer mesh to support basic telehealth functions regionally. While this is complex, it could improve disaster resilience (imagine a scenario where cloud connectivity is cut off due to a disaster; edges in a city could still coordinate via a local mesh ledger to ensure continuity of care and record-keeping until cloud is back).

Safety Assurance and Certification: As telehealth systems start handling more life-critical functions (like remote surgery), regulatory scrutiny on safety will increase. Future work involves creating rigorous models and simulations to verify that latency and reliability are within safe bounds for such applications. For instance, for telesurgery, one might need provable guarantees of max latency and fail-safe mechanisms if latency is exceeded (like an auto-pause of robotic motion). Research might develop formal verification methods for network control and failover logic in telehealth systems. The architecture may include redundant paths (like both wired and wireless concurrently for backup, or duplication of critical commands on two channels to ensure at least one gets through). Achieving a level of reliability close to traditional in-person or wired systems will be the goal for the most critical tasks.

6. Cross-domain Integration

Lastly, telehealth will integrate with other domains: smart home, smart city, emergency services. Future edge networks might coordinate with smart home hubs (the edge in your house not only monitors your health devices but also your environment like air quality, and alerts if something in environment triggers a health concern). At a city level, edge nodes might integrate with traffic management during an ambulance telehealth scenario to, for example, trigger traffic light control for the ambulance's route (some cities are exploring that). This cross-domain orchestration extends the concept of policy engines beyond health to community safety policies.

In conclusion, the future of edge-cloud telehealth systems is rich with possibilities. By embracing technologies like 5G slicing, confidential computing, and intelligent policy-driven management, we can create telehealth networks that are not only faster and safer, but also smarter and more adaptive. These advancements will bring us closer to a vision of healthcare that is **ubiquitously accessible, highly personalized, and uncompromising in quality**, regardless of physical distance between patient and provider. Our proposed architecture provides a solid foundation, and with the discussed future enhancements, it can evolve in tandem with technological progress to meet the needs of tomorrow's healthcare challenges.

Conclusion

Telehealth has transformed the way healthcare is delivered, breaking down geographical barriers and enabling continuous, remote patient care. However, to fully realize its potential, telehealth infrastructure must meet demanding requirements for **security, scalability, and low latency**. In this paper, we presented a comprehensive study and design of an edge-cloud hybrid network architecture aimed at fulfilling these requirements for telehealth systems. The architecture leverages the strengths of edge computing – bringing computation close to data sources for real-time responsiveness – in conjunction with the virtually unlimited resources of cloud computing for global scalability and data aggregation.

Through our detailed breakdown, we demonstrated how distributing computing to edge nodes can drastically reduce latency for critical telehealth applications (enabling near-real-time monitoring and intervention) and alleviate network bandwidth pressures by processing data locally [10] [6]. At the same time, cloud integration ensures that the system can scale to accommodate large user populations and heavy analytical tasks, as evidenced by industry examples like global telehealth platforms handling millions of patients with cloud back-ends [9]. Crucially, we did not treat security as an afterthought; instead, we wove a robust security framework into the architecture. This framework spans device-level authentication and encryption, secure communication channels (TLS/VPN) throughout, strict identity and access management, and privacy-preserving techniques. By employing end-to-end encryption and modern zero-trust principles, our design protects sensitive medical data whether it's on a wearable, traversing the network, or stored in the cloud [3] [6]. We also highlighted the need for strong key management and continuous monitoring to adapt to new threats, referencing recent research that underscores the effectiveness of such layered security in telehealth ecosystems [3] [3].

Our performance evaluation and discussion provided evidence that the hybrid approach can meet and exceed telehealth performance needs. We discussed a testbed scenario indicating significant latency reductions and improved reliability with edge involvement. Moreover, by analyzing the capabilities of GCP for hybrid deployments, we showed that the technological building blocks to implement this architecture are readily available and continually improving. Each platform offers unique tools like GCP's Anthos for portable edge services giving healthcare organizations flexibility in execution. The common thread is that all these platforms recognize the value of edge computing in low-latency, sensitive applications like healthcare, and have oriented their services to support such models. Our analysis confirms that choosing a hybrid design does not lock one into obscure technologies; on the contrary, it aligns with the direction of mainstream cloud offerings and standards.

In outlining future work, we painted a vision of telehealth networks that are even more integrated with upcoming technologies: leveraging 5G network slicing for guaranteed service quality, employing confidential computing to bolster patient privacy, and using intelligent policy engines and automation to manage complexity. The edge-cloud architecture we proposed is inherently agile – it can serve as the foundation upon which these future enhancements are built. For instance, as 5G MEC becomes ubiquitous, the edge layer of our architecture can naturally extend into those environments, further reducing latency for mobile telehealth [2]. As confidential computing matures, swapping in secure enclave processing at edge or cloud will enhance security without altering the overall design approach. Thus, our architecture is not a static end-point but a **flexible framework** designed to evolve.

In conclusion, the synthesis of our exploration is that a well-designed edge-cloud hybrid architecture is not only feasible but indeed vital for next-generation telehealth systems. It provides a balanced solution that meets the triad of security, scalability, and low-latency requirements in a way that neither edge nor cloud alone could achieve. By processing data at the edge when milliseconds matter, and aggregating data in the cloud when breadth and depth of analysis are needed, patients and providers get the best of both worlds: timely, responsive care as well as comprehensive, data-driven insights. The security measures ensure that this is done with unwavering commitment to patient privacy and data protection, a non-negotiable in healthcare.

The ideas and design principles discussed in this paper contribute to the growing body of knowledge on distributed healthcare systems and can inform the development of real-world implementations. As healthcare delivery continues to extend beyond traditional settings, architectures like the one proposed will play a crucial role in enabling equitable, effective, and safe telehealth services at scale. We hope that this work will encourage further research and collaboration between healthcare and IT professionals to refine these concepts and ultimately translate them into operational systems that benefit patients around the world.

References(10pt)

1. Rancea, A., Anghel, I., Cioara, T., et al., "Edge Computing in Healthcare: Innovations, Opportunities, and Challenges," *Future Internet*, vol.16, no.9, 2023. (Explores edge computing integration in healthcare, highlighting reduced latency and improved performance) (Edge Computing in Healthcare: Innovations, Opportunities, and Challenges)
2. Verizon Business, "Improving Patient Experience through On-Premises Telemedicine," *Verizon Enterprise Article*, 2023. (Discusses the need for edge computing in telemedicine to achieve real-time communication, with 5G integration) (Improving Patient Experience through On-Premises Telemedicine | Verizon Business)
3. Guo, Y., Guo, B., Guo, N., "Advancing security and privacy measures in telehealth IoT/Fog/Cloud ecosystems," *J. of Applied Biotechnology & Bioengineering*, vol.11, no.3, 2024. (Proposes a security model for telehealth across IoT, fog, cloud with end-to-end encryption and IDS) (Advancing security and privacy measures in telehealth IoT/Fog/Cloud ecosystems - MedCrave online)
4. Redington Group, "Edge Computing: A New Frontier in Telehealth," *Redington Blog*, 2023. (Highlights how local edge computing in telehealth can avoid latency issues and improve real-time decisions) (Revolutionize Telehealth with Edge Computing Solutions | Redington)
5. Williams, M., "Telemedicine needs the cloud and real-time relay as well," *Verizon Enterprise*, 2023. (Emphasizes that cloud capacity and edge real-time processing must be combined for telemedicine, citing robotic surgery latency needs) (Improving Patient Experience through On-Premises Telemedicine | Verizon Business)
6. Binariks, "How Edge Computing Improves Data Processing in Healthcare," *Binariks Blog*, 2024. (Describes benefits of edge in healthcare: low latency for time-sensitive data, reduced bandwidth, improved privacy by localizing data) (How Edge Computing Improves Data Processing in Healthcare)
7. NovelVista, "AWS Outposts vs. Local Zones vs. Wavelength: Edge Computing on AWS," *Blog*, 2023. (Provides use-cases of AWS hybrid services in healthcare, e.g. combining Outposts, Local Zones, Wavelength for a hospital chain to achieve HIPAA compliance and low latency) (AWS Outposts vs. AWS Local Zones vs. AWS Wavelength: Edge Computing on AWS)
8. Azure Team, "Microsoft Azure Stack and Edge for Healthcare," *Microsoft Azure Documentation/Case Studies*,

2023. (Details how Azure Stack and Azure Arc enable hybrid cloud in healthcare deployments, ensuring data residency and low latency processing on-premises – e.g., hospital use of Azure for telemedicine) (Azure's Digital Evolution: Cloud Telemedicine & Innovation)

9. Google Cloud, "Portal Telemedicina Case Study," *Google Cloud Customers*, 2022. (Case study of a telehealth provider using Google Cloud IoT and AI to serve 30+ million patients, illustrating cloud scalability and the use of edge gateways for data collection) (Portal Telemedicina Case Study | Google Cloud)
10. Lakshminarayanan, V., et al., "Health Care Equity Through Intelligent Edge Computing and AR/VR: A Systematic Review," *Frontiers in Digital Health*, 2023. (Survey of edge computing in healthcare, noting that processing at the edge offers low latency and privacy benefits, and discussing challenges of edge security and data integration) (Health Care Equity Through Intelligent Edge Computing and Augmented Reality/Virtual Reality: A Systematic Review - PMC)
11. Alex Flores, "Maximizing AI Deployment Value in Healthcare Requires a Hybrid Edge-to-Cloud Strategy"(Healthcare organizations must also optimize their computing resources to support innovative clinical workflows.)(Maximizing AI Deployment Value in Healthcare Requires a Hybrid Edge-to-Cloud Strategy | HealthTech Magazine)
12. Q. He et al., "Telemedicine Monitoring System Based on Fog/Edge Computing: A Survey" in IEEE Transactions on Services Computing, vol. 18, no. 01, pp. 479-498, Jan.-Feb. 2025, doi: 10.1109/TSC.2024.3506473.keywords: {Edge computing;Cloud computing;Telemedicine;Medical services;Surveys;Real-time systems;Internet of Things;Reliability;Data privacy;Low latency communication}URL:https://doi.ieeecomputersociety.org/10.1109/TSC.2024.3506473
13. Aledhari, Mohammed & Razzak, Rehma & Qolomany, Basheer & Al-Fuqaha, Ala & Saeed, Fahad. (2022). Biomedical IoT: Enabling Technologies, Architectural Elements, Challenges, and Future Directions. IEEE Access. 10. 10.1109/ACCESS.2022.3159235.
14. Lakshminarayanan V, Ravikumar A, Sriraman H, Alla S, Chattu VK. Health Care Equity Through Intelligent Edge Computing and Augmented Reality/Virtual Reality: A Systematic Review. J Multidiscip Healthc. 2023 Sep 21;16:2839-2859. doi: 10.2147/JMDH.S419923. PMID: 37753339; PMCID: PMC10519219.
15. Janakiram MSV, "Google Finally Gets The Edge Computing Strategy Right With Distributed Cloud Edge"(https://www.linkedin.com/pulse/google-finally-gets-edge-computing-strategy-right-distributed-msv/)