

Arquitetura Multicamadas para Coleta e Análise de Dados de Saúde em Tempo Real em Ambientes Externos, Integrando Fog Computing e Cloud Computing

Juan Felipe Souza Oliveira
Instituto Militar de Engenharia
Rio de Janeiro, Brasil
oliveira.juan@ime.eb.br

Paulo Cesar Salgado Vidal
Instituto Militar de Engenharia
Rio de Janeiro, Brasil
vidal@ime.eb.br

Ronaldo Moreira Salles
CIICESI, ESTG, Instituto Politécnico do Porto
Porto, Portugal
rmo@estg.ipp.pt

Marcelo Quesado Filgueiras
Universidade Federal de Juiz de Fora
Juiz de Fora, Brasil
mquesado@uol.com.br

ABSTRACT

A multilayer architecture was developed for real-time health data collection and processing, optimized for outdoor environments with high population density and significant network interferences, integrating fog and cloud computing. With the increasing adoption of the Internet of Things (IoT) and Wireless Body Area Networks (WBANs) using smartbands, continuous health monitoring generates vast amounts of data that require efficient processing and reliable transmission. Traditional cloud-based solutions, while scalable, often face high latency and data integrity challenges in unstable network conditions. By leveraging fog computing, the developed architecture performs data preprocessing at the network edge, reducing dependency on cloud connectivity and enhancing system responsiveness. Real-world tests were conducted in complex environments such as football stadiums, beaches, and metro systems, with varying network conditions (5G, 4G, 3G). The architecture consistently achieved over 96% success in packet delivery and significantly reduced latency compared to cloud-only solutions. These results highlight the architecture's resilience and effectiveness for real-time health monitoring, ensuring data integrity and low response times in high-density, interference-prone environments.

KEYWORDS

internet das coisas, monitoramento remoto de saúde, computação em névoa

1 INTRODUÇÃO

O crescimento exponencial dos sistemas de saúde e o aumento da expectativa de vida da população mundial trazem consigo diversos desafios, especialmente no que diz respeito ao manejo e controle de doenças crônicas e epidêmicas que ameaçam a saúde pública [10]. Monitorar continuamente os parâmetros de saúde dos pacientes, tais como frequência cardíaca, níveis de oxigenação no sangue, índice

glicêmico e peso, é essencial para enfrentar essas questões. No entanto, esse monitoramento exige alta integridade e confiabilidade dos dados transmitidos por redes móveis e pela Internet [11].

A Internet das Coisas (IoT) emergiu como uma solução promissora para esses desafios, integrando sensores, dispositivos vestíveis, bancos de dados e computação em nuvem (*cloud computing*) [24]. Dispositivos vestíveis sem fio, como pulseiras inteligentes (*smartbands*), permitem o monitoramento de pacientes fora das unidades de saúde, promovendo economia de recursos humanos e materiais [17], além de ampliar o acompanhamento integral de pacientes que necessitam de um diagnóstico ágil de suas patologias, principalmente aquelas com elevados índices de incapacitação e mortalidade [18][2].

A computação em nuvem oferece vantagens, como escalabilidade, capacidade de processamento de dados e economia de custos operacionais. Permite a alocação dinâmica de recursos computacionais conforme a demanda, otimizando o uso de servidores e infraestrutura [12]. Além disso, facilita a implementação de soluções de banco de dados de alta disponibilidade, proporcionando colaboração em tempo real e mobilidade, o que facilita o acesso a dados e aplicativos de qualquer lugar e a qualquer hora [23].

No entanto, enfrenta algumas limitações, como a dependência de uma conexão estável à Internet, maior latência devido à distância entre o dispositivo do usuário e os data centers na nuvem [16], e possíveis vulnerabilidades de segurança durante a transmissão dos dados [25].

Para mitigar essas limitações, a computação em névoa surge como uma evolução da computação distribuída, com o objetivo de superar as limitações dos modelos atuais baseados em computação em nuvem [16][6]. É possível processar os dados e obter recursos de armazenamento e comunicação na borda da rede, mais próxima dos sensores e usuários. Isso pode proporcionar um monitoramento contínuo e em tempo real, disparos alertas com menor latência, mesmo em condições de falha na rede [26], bem como pré processamento de dados antes de serem enviados para a camada de nuvem [6]. Essa capacidade de processamento e análise descentralizados são particularmente vantajosos para aplicações de saúde que requerem respostas rápidas e confiáveis [15].

Alguns trabalhos realizaram implementações de computação em névoa e nuvem para o monitoramento de saúde, contudo, poucos [3][1] exploraram testes e avaliações de integridade dos dados coletados e latência em ambientes externos com alta densidade populacional e interferências, como praias, estádios de futebol e metrô. Esses ambientes apresentam desafios adicionais devido à alta quantidade de dispositivos conectados e à instabilidade das redes.

O trabalho em [19] propôs uma arquitetura de monitoramento de sinais vitais utilizando computação em névoa, chamada CEN. A arquitetura é composta de 3 (três) camadas (sensores, névoa e nuvem) e emprega o protocolo de comunicação *Message Queuing Telemetry Transport (MQTT)*. O protocolo *MQTT* é caracterizado por ser mais leve (em termos de processamento) e eficiente e ideal para transmissão de dados em tempo real em redes com recursos limitados e instáveis [20], sendo avaliado nos cenários do Estádio do Maracanã e Praia de Copacabana.

O presente artigo se diferencia de [19] por descrever com mais detalhes a implementação da arquitetura e por realizar testes em outros ambientes externos na cidade do Rio de Janeiro, tais como o metrô, o Estádio Olímpico Nilton Santos (Engenhão) e o trecho da Baía de Guanabara (entre as cidades de Niterói e Rio de Janeiro). Esses testes em diferentes cenários tiveram como objetivo demonstrar a eficiência da arquitetura em condições de instabilidade de rede.

O artigo está organizado da seguinte forma: Na Seção 2, discutimos os desafios técnicos e operacionais da computação em nuvem e névoa na área médica. Na Seção 3, revisamos trabalhos relacionados e comparamos com nossa abordagem. A Seção 4 detalha a arquitetura desenvolvida e suas características. Na Seção 5, abordamos a avaliação das implementações e cenários de testes. Na Seção 6, apresentamos os resultados dos experimentos, incluindo desempenho em cenários externos, testes de latência e atrasos na emissão de alertas. Finalmente, na Seção 7, concluímos o artigo e sugerimos trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 Computação em Nuvem na Área Médica

Aplicações na área da saúde desenvolvidas e baseadas em computação em nuvem possuem alta capacidade de armazenamento e poder computacional, logo, têm sido conhecidas como formas eficientes de transferir, processar e armazenar dados [22]. No entanto, existem algumas deficiências dos modelos atuais [6], tais como:

- **Infraestrutura de rede:** A falta de conectividade com servidores, dados ou aplicações hospedadas na nuvem devido a falhas nas redes de comunicação, como as redes móveis, representa um grande obstáculo. Em contextos médicos, onde o acesso contínuo a informações é essencial para decisões rápidas, a interrupção de rede pode causar atrasos prejudiciais no atendimento.
- **Largura de banda:** O grande volume de tráfego de dados pode levar os provedores de nuvem a aplicar restrições na largura de banda para garantir o funcionamento adequado da rede. Isso pode afetar negativamente o desempenho de aplicações críticas na área da saúde, como monitoramento que exigem transferência de grandes volumes de dados em tempo real. Com restrições de largura de banda, os tempos

de resposta podem ser significativamente aumentados, dificultando o acesso rápido e eficaz às informações necessárias para decisões imediatas.

- **Confiabilidade:** Problemas na nuvem podem resultar em várias implicações. A integridade dos dados é vital para garantir a precisão das informações e a continuidade dos serviços. Em situações de falha, pode ocorrer perda ou corrupção de dados, afetando diretamente a consistência de prontuários médicos e outros registros essenciais. Isso pode resultar em diagnósticos incorretos ou atrasos no tratamento, comprometendo a segurança e o bem-estar dos pacientes.
- **Tempo de resposta:** Aplicações médicas, como sistemas de monitoramento de pacientes, exigem tempos de resposta extremamente rápidos para alertar os profissionais de saúde sobre mudanças críticas no estado do paciente. A latência na comunicação com a nuvem pode causar atrasos na detecção e resposta a essas mudanças, potencialmente comprometendo a capacidade dos profissionais de saúde de intervir de maneira oportuna e eficaz.

Tendo em vista os aspectos observados, modelos de *IoT* utilizando apenas a camada de computação em nuvem não necessariamente oferecem a solução mais viável para aplicativos críticos [3] e enfrentamento dos desafios de *IoT* atrelados à área médica [22]. Nesse contexto, computação em névoa surge como um paradigma de computação distribuída que estende a capacidade da nuvem para dispositivos localizados próximos aos usuários ou no limite da rede, como sensores, câmeras e redes corporais sem fio

2.2 Computação em Névoa na Área Médica

A integração de tecnologias de computação em névoa com dispositivos de saúde, como *smartbands* e *smartphones*, apresenta desafios técnicos e operacionais que devem ser superados para garantir a eficácia e segurança dos sistemas de monitoramento de saúde [25]. A computação em névoa oferece vantagens significativas, como processamento descentralizado e melhoria na capacidade de resposta em tempo real, mas também traz desafios para uma implementação bem-sucedida [6][1][25], tais como:

- **Segurança e Privacidade dos Dados de Saúde:** A proteção dos dados de saúde é crucial devido à sua sensibilidade. A comunicação entre a *smartband*, o *smartphone* (nó de névoa) e a nuvem envolve múltiplas etapas suscetíveis a vulnerabilidades, exigindo criptografia ponta a ponta e autenticação robusta para garantir a integridade dos dados.
- **Gestão de Conectividade e Interferências:** Ambientes densamente povoados, como praias, estádios e metrô, apresentam desafios devido à alta densidade de dispositivos conectados, bem como o relevo do terreno, resulta em congestionamentos e interferências na rede. Utilizar um protocolo de comunicação leve como o *MQTT* pode mitigar esses problemas.
- **Complexidade de Implementação e Integração:** Integrar *smartbands*, *smartphones* e camada de computação em nuvem em uma arquitetura envolvendo a computação em névoa, necessita da coordenação de múltiplos componentes heterogêneos. A interoperabilidade entre dispositivos e plataformas,

como *Android* e *Apple*, é essencial, assim como o desenvolvimento de aplicações e *plugins* especializadas para gerenciar a comunicação entre dispositivos e a camada de computação em nuvem.

Em resumo, a integração de dispositivos de saúde com a computação em *névoa* e *nuvem* enfrenta desafios críticos: segurança e privacidade dos dados, gestão de conectividade em ambientes externos com vários dispositivos conectados, e a complexidade de integração de sistemas heterogêneos. Superar esses desafios é essencial para uma arquitetura de multicamadas, combinando sensores, camada de *névoa* e *nuvem*.

3 TRABALHOS RELACIONADOS

Os trabalhos relacionados nesta área de pesquisa exploraram os conceitos de latência, tolerância a falhas [13], simulação de coleta e coleta de dados reais em sistemas de monitoramento de parâmetros de saúde [18]. Esses estudos investigaram abordagens para minimizar a latência na transmissão e processamento de dados, bem como propuseram técnicas para lidar com falhas e instabilidades nas redes de comunicações móveis [5][8]. Além disso, utilizaram simulações para avaliar o desempenho dos sistemas em diferentes cenários e também realizaram coletas de dados reais para validarem suas propostas [21].

A revisão dos trabalhos relacionados abordou critérios relevantes alinhados com a arquitetura desenvolvida neste artigo, tais como:

- **Latência:** Garantir que o tempo de transmissão e processamento dos dados seja minimizado para proporcionar respostas rápidas e eficientes. Isso exige a otimização da latência tanto na computação em *névoa* quanto na *nuvem*, de forma a manter a agilidade do sistema.
- **Tolerância a falhas:** Desenvolver arquiteturas capazes de suportar falhas em dispositivos, redes ou processos, mantendo a operação do sistema sem interrupções. Isso assegura a continuidade do serviço e a confiança, mesmo em condições adversas.
- **Smartphone como nó da camada de névoa (Fog Node):** Os *smartphones* desempenham um papel chave na arquitetura de computação em *névoa*, atuando como nós periféricos da rede. Com seu alto poder de processamento e armazenamento, eles podem realizar tarefas computacionais localmente, reduzindo a latência e se conectando de forma eficiente a redes móveis, *Wi-Fi* e outros dispositivos próximos.
- **Diferentes protocolos de comunicação de rede:** Aplicação de protocolos, como *HTTP*, *MQTT* e *CoAP*, no contexto da saúde, explorando diferentes abordagens. Esses protocolos oferecem suporte à troca eficiente de dados clínicos, ao monitoramento remoto de pacientes e à integração com redes corporais sem fio, promovendo uma comunicação mais eficaz entre dispositivos.
- **Coleta de dados:** validar e calibrar os sistemas de monitoramento, permitindo verificar confiabilidade e eficácia do sistema, e assim otimizar o sistema como um todo com base nas condições reais de uso.

Os estudos comparativos abordaram os desafios enfrentados em ambientes internos e externos, considerando a presença ou ausência de redes de comunicação móveis, como 2G, 3G, 4G e 5G, e a

utilização de *Wi-Fi* [22][15]. Ambientes internos foram identificados como oferecendo infraestrutura de rede mais estável, facilitando a comunicação entre dispositivos de monitoramento de saúde. Em contraste, ambientes externos apresentam maior variabilidade e desafios de conectividade.

Soluções foram propostas para lidar com a falta de rede e instabilidades em ambientes externos [21], incluindo armazenamento temporário de dados, tecnologias de armazenamento local e redes *ad hoc* [9]. Essas estratégias visam garantir a confiabilidade da coleta de dados em ambientes desafiadores. Em suma, os estudos destacaram a importância de considerar as características de cada ambiente, a presença ou ausência de redes móveis e *Wi-Fi*, e as diferentes gerações de redes móveis no projeto de sistemas de monitoramento de parâmetros de saúde em diferentes contextos.

As tecnologias e ferramentas computacionais utilizadas em trabalhos anteriores [20][21], como banco de dados não-relacionais *NoSQL*, sistemas de mensageria e o padrão de projeto em arquitetura de software *Design Pattern*, *Transactional Outbox*, contribuíram para um aprofundamento e implementação da arquitetura desenvolvida. Essas contribuições técnicas fornecem melhorias significativas, permitindo uma coleta de dados mais eficiente, confiável e escalável.

Os bancos de dados *NoSQL* permitem que os sistemas de monitoramento de saúde lidem de forma flexível e eficiente com grandes volumes de dados, especialmente dados heterogêneos e não estruturados. Eles possibilitam rápida inserção e consulta de dados em tempo real, essencial para o monitoramento contínuo de pacientes [7][21]. É observado também a capacidade de ampliar a escalabilidade horizontal, permitindo a expansão do sistema para gerenciar mais aplicações e dados coletados [4], garantindo uma resposta rápida e eficaz às necessidades dos pacientes.

Em relação a ferramentas de sistemas de mensageria e *streaming* de dados distribuídos para lidar com fluxos contínuos de dados em tempo real, é empregada a comunicação assíncrona entre os componentes do sistema, tornando a troca de informações mais flexível e escalável [14], desempenhando papel fundamental no monitoramento em ambientes externos, onde as condições de rede podem ser instáveis. Além disso, do padrão *Transactional Outbox* garante a consistência e a atomicidade nas operações de envio de mensagens [22][21], evitando perdas ou duplicações de dados durante a comunicação assíncrona.

Por fim, para o tráfego dos dados para a camada de *nuvem*, estudos recentes demonstram que o protocolo de comunicação *Message Queuing Telemetry Transport (MQTT)* apresenta alta eficiência e tolerância a falhas de conexão [3][5], permitindo reconexão automática quando há interrupção ou alteração na conectividade, sendo possível adaptar-se a diferentes níveis de qualidade de rede, incluindo as de baixa largura de banda e conexões instáveis, como as de 2G e 3G, ou em situações de sobrecarga da rede devido a vários dispositivos móveis conectados simultaneamente, otimizando o tráfego dos dados com base nas condições da rede [22][15].

Apesar dos avanços e contribuições significativas dos trabalhos analisados, algumas limitações foram identificadas. Primeiramente, muitos estudos não abordaram de forma abrangente a integração entre diferentes tecnologias de comunicação, como *Wi-Fi* e redes móveis de diferentes gerações (2G, 3G, 4G e 5G), resultando em possíveis lacunas na continuidade do monitoramento em ambientes

com variabilidade de conectividade. Embora alguns trabalhos tenham proposto soluções para tolerância a falhas e armazenamento temporário de dados, a implementação dessas soluções em cenários reais ainda carece de testes extensivos e validações práticas. A maioria das abordagens focou-se em ambientes internos, onde a estabilidade da rede é maior, deixando uma lacuna significativa em estudos voltados para ambientes externos com condições adversas de conectividade.

Outra limitação observada é a subutilização do poder computacional dos *smartphones* como nós da camada de névoa. Muitos estudos não exploraram totalmente o potencial destes dispositivos para executar tarefas computacionais na borda da rede, o que pode reduzir a latência e melhorar a eficiência do sistema. Essas limitações destacam a necessidade de abordagens mais integradas e soluções que possam ser aplicadas em uma variedade de cenários de monitoramento de saúde.

O presente artigo se diferencia ao apresentar uma arquitetura que será testada em ambientes externos com condições adversas de conectividade, como praias, estádios de futebol, e transporte público como metrô, proporcionando uma avaliação mais abrangente e prática.

A arquitetura desenvolvida utiliza poder computacional dos *smartphones* como nós da camada de névoa (*fog nodes*), onde são realizados o pré-processamento, a criptografia e a anonimização dos dados para conformidade com a LGPD (Lei Geral de Proteção de Dados).

Por fim, a aplicação de bancos de dados *NoSQL* e o uso do protocolo de *MQTT* para fluxo contínuo de dados em tempo real de dados, apresentam uma perspectiva de superar os desafios de escalabilidade e eficiência na transmissão de dados. Essas abordagens visam melhorar a eficiência, reduzir a latência e aumentar a resiliência do sistema, representando uma contribuição significativa para o campo do monitoramento de saúde em ambientes externos.

4 ARQUITETURA DESENVOLVIDA

4.1 Concepção da Arquitetura

As etapas de desenvolvimento e implementação da arquitetura estão integradas aos princípios fundamentais da Engenharia de Software e à garantia da Qualidade de Software. Cada fase, desde o levantamento bibliográfico até os testes em ambientes reais, demonstra uma abordagem metodológica e técnica para garantir um completo desenvolvimento da arquitetura. O processo inclui: a) levantamento bibliográfico, b) definição de requisitos funcionais e não funcionais, c) seleção de tecnologias e *frameworks*, d) testes preliminares de variações de redes de comunicações em ambientes externos, e) desenvolvimento da arquitetura, f) testes da arquitetura em ambientes externos, g) avaliação de desempenho da arquitetura.

A arquitetura de monitoramento de saúde inclui três camadas principais: sensores, computação em névoa e computação em nuvem. Os sensores, como *smartbands*, coletam dados vitais em tempo real. A computação em névoa, utilizando *smartphones* como nós de névoa, realiza o pré-processamento, criptografia, anonimização e persistência dos dados em caso de instabilidade de conexão, além de transmitir dados para a nuvem via *MQTT*. A computação em nuvem é responsável pelo armazenamento, análise, geração de

dashboards de acompanhamento e emissão de alertas, permitindo monitoramento em tempo real.

A partir da conexão *Bluetooth* entre a *smartband* e o *smartphone* do paciente, será possível realizar a coleta contínua de parâmetros de saúde, fornecendo informações precisas e atualizadas. Esta arquitetura viabiliza um monitoramento mais eficaz e personalizado.

Ao processar os dados próximos à fonte por meio da computação em névoa, a arquitetura poderá garantir uma coleta e transmissão de dados mais eficiente e ágil. Isso pode solucionar desafios relacionados à geração de dados eletrônicos confiáveis, atendimento a requisitos temporais e tolerância a falhas, coletando parâmetros de saúde como frequência cardíaca, número de passos e geolocalização do paciente.

Os dados coletados e processados poderão ser utilizados para obter uma visão clara da condição de saúde dos pacientes e serão enviados periodicamente para servidores em nuvem. No contexto de *e-Health*, ambulâncias, enfermeiros e médicos poderão acessar remotamente essas informações para avaliar a condição de saúde dos pacientes monitorados.

4.2 Implementação

4.2.1 Camada de Computação em névoa. Na Figura 1, é apresentado o diagrama de componentes e ferramentas da arquitetura de computação em névoa, incluindo a camada de sensores e o aplicativo móvel (*Publisher API*) desenvolvido para *Android* e *Apple*, que se comunica com a nuvem. Utilizando a conexão *Bluetooth* entre a *smartband* e o *smartphone* do paciente, são coletados continuamente parâmetros de saúde como frequência cardíaca, número de passos e geolocalização.

Na implementação desta camada, o *smartphone* atua como nó da névoa, realizando diversas funções críticas. Primeiramente, ele realiza o pré-processamento dos dados coletados. No caso de instabilidade de conexão com a nuvem, o *smartphone* utiliza um banco de dados local para persistir temporariamente os dados. Isso garante que nenhuma informação seja perdida durante interrupções na conectividade.

Os dados processados são enviados para a nuvem via protocolo *MQTT*, que assegura a entrega eficiente e resiliente das informações. Na camada de névoa, além de receber os dados dos sensores, ocorre o processo de *ETL* (*Extract, Transform, Load*), que inclui a extração dos dados brutos, a aplicação de filtros e o armazenamento local. Esse processo permite a seleção e processamento apenas dos dados relevantes, reduzindo a quantidade de informações a serem enviadas para a nuvem e otimizando o consumo de recursos.

Para gerenciar a persistência e a transmissão dos dados, implementamos o padrão de projeto *Transactional Outbox*. Este padrão permite que, em caso de falha ao se conectar ao *broker MQTT* na nuvem, as mensagens sejam armazenadas localmente. Quando a conexão é restabelecida, o banco de dados local é consultado e os dados armazenados são reenviados automaticamente ao *broker MQTT*. Isso assegura que a integridade dos dados seja mantida e que todas as informações sejam corretamente processadas e transmitidas, mesmo em condições adversas de conectividade.

4.2.2 Camada de Computação em Nuvem. Na Figura 2, é apresentado o diagrama de componentes e ferramentas da arquitetura de computação em nuvem, que complementa a camada de névoa.

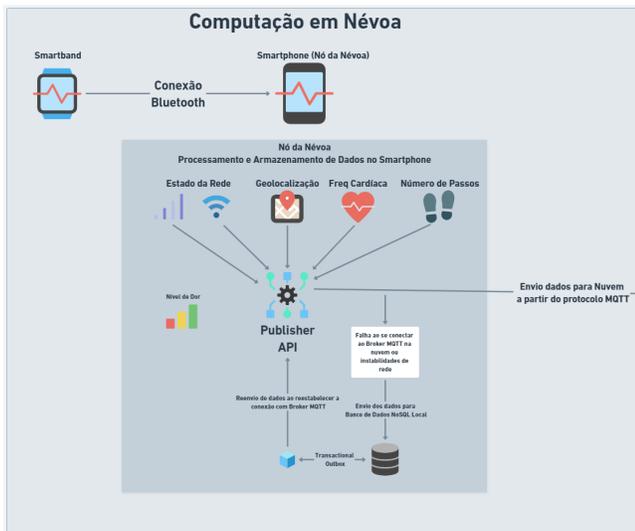


Figure 1: Diagrama de Componentes da Camada de Computação em névoa (Autor)

Esta camada, hospedada na AWS (*Amazon Web Services*), oferece uma infraestrutura escalável e segura, garantindo conformidade com a LGPD ao implementar políticas rigorosas de segurança e privacidade.

Os dados processados e transmitidos via protocolo *MQTT* pelo *smartphone* são enviados para um *broker MQTT*, que atua como intermediário na comunicação entre a camada de névoa e a nuvem. Em caso de desconexão do *smartphone*, o *broker* envia logs para um banco de dados *NoSQL* através do *Amazon SNS (Simple Notification Service)* e dispara *Will Messages* — mensagens configuradas para serem enviadas automaticamente quando ocorre uma desconexão — para a *Subscriber API*.

A *Subscriber API*, implementada em *Java* com o framework *Spring*, consome as mensagens do *broker MQTT*. O *Spring* facilita a criação de aplicações *Java*, proporcionando uma infraestrutura robusta para gerenciar componentes, configurar funcionalidades e integrações *broker MQTT*, *RabbitMQ* e bancos de dados. A *Subscriber API* enfileira as mensagens no *RabbitMQ*, um sistema de mensageria que permite a comunicação assíncrona e escalável entre os componentes do sistema, e armazena-as no banco de dados *NoSQL*. Para garantir a anonimização, a nuvem recebe apenas um identificador único (ID) associado aos dados coletados, sem informações pessoalmente identificáveis dos pacientes.

Com a utilização do *Prometheus*, que é um sistema de monitoramento e alerta que coleta e armazena métricas em séries temporais, permitindo flexibilidade na coleta e consulta de dados, é possível monitorar a saúde do sistema e a integridade dos dados. O *Prometheus* coleta dados de vários componentes do sistema, incluindo o *broker MQTT*, a *Subscriber API* e o banco de dados *NoSQL*, e armazena essas métricas em uma base de dados otimizada para consultas temporais.

O *Grafana* é uma plataforma de visualização e análise de código aberto que se integra ao *Prometheus* para gerar dashboards interativos e painéis de controle. Esses dashboards permitem o acompanhamento em tempo real dos parâmetros de saúde dos pacientes, disparam alertas quando métricas específicas atingem determinados limites e fornecem uma interface de usuário (*User Interface*) acessível remotamente por profissionais de saúde. O *Grafana* permite que os usuários criem visualizações customizadas dos dados coletados pelo *Prometheus*, facilitando a detecção de anomalias e a tomada de decisões informadas.

Por fim, quando o *broker MQTT* detectar uma desconexão com o *smartphone*, ele poderá disparar *Will Messages* para a *Subscriber API*, indicando a desconexão, o nível de bateria do *smartphone*, a hora e o local. A *Subscriber API* poderá então enviar essas mensagens para o *Telegram* para notificar os responsáveis. Esses logs de eventos poderão ser importantes para a pesquisa, pois permitirão o monitoramento da continuidade e da qualidade da conexão nos ambientes externos. Isso possibilitará a identificação rápida de problemas de conectividade e a implementação de ações corretivas imediatas.

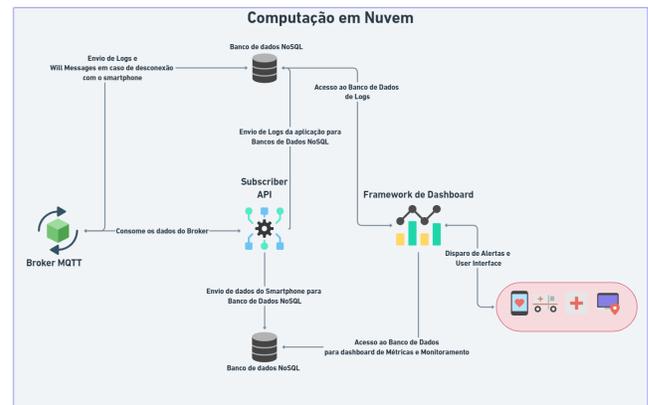


Figure 2: Diagrama de Componentes da Camada de Computação em névoa (Autor)

5 AVALIAÇÃO

Além da arquitetura principal desenvolvida em camadas (névoa e nuvem), chamada de implementação com computação em névoa (CEN), também foi desenvolvida uma implementação secundária chamada de implementação sem computação em névoa (SCEN). A implementação SCEN utiliza apenas a camada de computação em nuvem, onde o *smartphone* envia os dados diretamente para a nuvem através do protocolo de comunicação *HTTP*, sem a utilização da camada de computação em névoa.

Realizaremos a avaliação da arquitetura desenvolvida, comparando as duas implementações distintas descritas anteriormente. Esta avaliação visa identificar as vantagens e limitações de cada abordagem, especialmente em termos de confiabilidade e eficiência na transmissão de dados de saúde.

Por fim, a avaliação incluirá a análise da arquitetura como um todo, considerando aspectos como latência, emissão de alertas e capacidade de operar em diferentes ambientes com variabilidade

de conectividade. Os critérios de avaliação incluirão a estabilidade da conexão, a taxa de transferência de dados, a perda de pacotes e a eficácia na emissão de alertas em tempo real.

5.1 Arquitetura Principal e Implementação Secundária

A utilização do protocolo *HTTP* apresenta limitações significativas em comparação com o *MQTT*. O *HTTP* é um protocolo baseado em requisições e respostas, o que significa que a comunicação é unidirecional e somente ocorre quando o *smartphone* envia uma solicitação e a nuvem responde. Isso pode resultar em maior latência e menor eficiência na transmissão de dados em tempo real, pois cada comunicação requer o estabelecimento de uma nova conexão [22]. Além disso, o *HTTP* não possui mecanismos internos para garantir a entrega dos dados, o que pode levar à perda de informações em condições de rede instáveis ou intermitentes [20]. Essas limitações podem impactar negativamente a confiabilidade e a eficiência da transmissão de dados críticos de saúde.

Em contraste, o *MQTT* é projetado para comunicação bidirecional contínua, permitindo que os dados sejam enviados e recebidos em tempo real com menor latência. O *MQTT* oferece garantias de entrega de mensagens através de diferentes níveis de qualidade de serviço (QoS), essenciais para a confiabilidade dos dados de saúde em ambientes com conectividade instável [5].

Na arquitetura, utilizamos o nível de QoS 1, que garante que uma mensagem seja entregue pelo menos uma vez, com confirmação de recebimento. Este nível de serviço equilibra confiabilidade e eficiência, garantindo a entrega dos pacotes de dados mesmo em caso de falhas temporárias de rede, sem a sobrecarga de múltiplas transmissões redundantes [5]. Isso mantém uma alta taxa de entrega de mensagens com baixa latência.

5.2 Critérios de Avaliação

A comparação entre a Computação em Névoa (CEN) e a Sem Computação em Névoa (SCEN), com foco na conexão com o *Broker MQTT* abordará limitações identificadas em trabalhos anteriores, como a falta de integração entre diferentes tecnologias de comunicação e a subutilização do poder computacional dos *smartphones*.

A arquitetura visa superar essas limitações utilizando *smartphones* como nós da camada de névoa (*fog nodes*). A aplicação de bancos de dados *NoSQL* e o uso do protocolo *MQTT* para fluxo contínuo de dados em tempo real são elementos centrais dessa abordagem.

A comparação entre as implementações CEN e SCEN será conduzida com base nos seguintes critérios:

- **Estabilidade da Conexão:** Avaliar a frequência de desconexões e a capacidade de reconexão automática.
- **Latência:** Medir o tempo de transmissão dos dados desde a coleta até a persistência dos dados nos bancos de dados presentes na camada de computação em nuvem.
- **Taxa de Transferência de Dados:** Avaliar a quantidade de dados transmitidos ao longo dos experimentos.
- **Perda de Pacotes:** Monitorar os dados perdidos durante a transmissão.

5.3 Cenários de Teste

As coletas e aferições de dados foram realizadas em três cenários distintos usando o software *Open Signal*, que testa sinal de rede, incluindo *download*, *upload* e latência, e permite detectar a presença e qualidade de sinal e identificar a performance da rede.

- **Metrô do Rio de Janeiro, Linha 1:** Observou-se variações constantes na qualidade do sinal, com áreas de ausência de conectividade e sinal fraco a moderado. Estações subterâneas apresentaram alta latência e desconexões frequentes, comprometendo a continuidade do monitoramento de saúde em tempo real.
- **Estádio Olímpico Nilton Santos (Engenhão), Rio de Janeiro - RJ:** Em dias de jogos, a rede sofre sobrecarga devido ao grande número de dispositivos conectados, resultando em instabilidade, atrasos e perdas de dados, impactando negativamente o monitoramento de saúde.
- **Baía de Guanabara, Rio de Janeiro - RJ:** Durante o deslocamento da barca entre as cidades do Rio de Janeiro e Niterói, o trajeto apresentou variações na qualidade do sinal devido à interferência do ambiente marinho e à adaptação à mudança de áreas de cobertura, resultando em aumento da latência e desconexões ocasionais.

Estes cenários de teste destacam os desafios significativos na manutenção de uma conexão de rede estável e confiável em ambientes urbanos e de alta densidade populacional. A análise desses dados é essencial para avaliar a eficácia das implementações CEN e SCEN em diferentes condições de conectividade e para identificar áreas potenciais para melhorias na transmissão de dados críticos de saúde.

6 RESULTADOS

Esta seção analisa os resultados de dois experimentos em três cenários de teste como mostrado na Figura 3. O primeiro experimento avaliou o desempenho das arquiteturas SCEN e CEN no Metrô e no Estádio Olímpico Nilton Santos (Engenhão), focando na eficiência em lidar com intermitências de rede. Durante os testes, pacotes de dados foram enviados a cada segundo para monitorar informações em tempo real. O segundo experimento testou a latência e emissão de alertas no Engenhão e no deslocamento de barca entre Rio de Janeiro e Niterói, analisando a comunicação em cenários de alta densidade e variação de rede.

6.1 Desempenho das Arquiteturas CEN e SCEN no Metrô

Durante o experimento no metrô, ambas as arquiteturas foram utilizadas simultaneamente, e identificamos 12 intermitências na transmissão, variando entre 10 segundos e 2 minutos, ao longo de 90 minutos, totalizando o envio de 5400 pacotes de dados. Essas intermitências causaram perda de pacotes. A Tabela 1 apresenta os resultados, destacando o desempenho de cada arquitetura frente às intermitências.

Essa abordagem SCEN, embora simples via protocolo HTTP, pode introduzir latência adicional e é mais susceptível a perda de dados em casos de intermitência na conexão, devido à necessidade de estabelecer uma nova conexão para cada transação de dados. Os

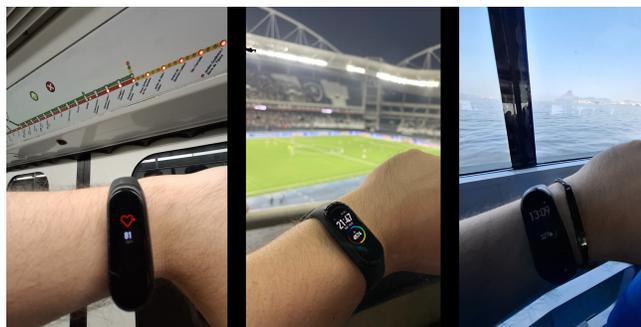


Figure 3: Coleta de dados nos cenários de teste: (a) Metrô, (b) Estádio de futebol, e (c) Baía de Guanabara, utilizando a pulseira *Xiaomi Mi Smart Band 4* durante a execução dos experimentos. (Autor)

Table 1: Comparação de Métricas de Desempenho no Metrô

| Métrica | SCEN | CEN |
|---------------------|-------|-------|
| Total de Pacotes | 5400 | 5400 |
| Pacotes Perdidos | 1680 | 176 |
| Pacotes de Sucesso | 3720 | 5224 |
| Taxa de Sucesso (%) | 68.89 | 96.73 |
| Taxa de Falha (%) | 31.11 | 3.27 |

valores observados na tabela para SCEN refletem essa dinâmica, com uma taxa de sucesso de 68.89% e uma taxa de falha de 31.11%, evidenciando desafios potenciais em termos de latência e perda de dados durante interrupções na conectividade.

Por outro lado, a arquitetura CEN ao utilizar o protocolo *MQTT*, suportando conexões persistentes e permitindo o uso do padrão *Transactional Outbox* para registrar os dados coletados localmente antes de serem enviadas à nuvem em caso de desconexão e intermitências da rede. Essa abordagem oferece uma vantagem em ambientes externos, onde a conectividade demonstrou ser instável. Os dados da tabela para CEN mostram uma taxa de sucesso de 96.73% e uma baixa taxa de falha de 3.27%, destacando a eficiência da arquitetura CEN em garantir uma resposta rápida a eventos críticos com mínima sobrecarga de rede e menor susceptibilidade a perda de dados.

6.2 Desempenho da Arquiteturas SCEN e CEN no Estádio Olímpico Nilton Santos (Engenhão)

Após a avaliação no metrô, um experimento semelhante foi realizado no estádio, durante o clássico entre Vasco e Fluminense, em 10/08/2024, com um público de 20.003 torcedores (Figura 3). Durante as duas horas do evento, foram enviados 7200 pacotes de dados. Observou-se uma degradação significativa na qualidade das redes móveis devido à sobrecarga de dispositivos conectados, resultando em alta latência, taxas de transferência reduzidas e perda de pacotes. A Tabela 2 compara o desempenho de cada arquitetura nesse cenário.

Table 2: Comparação de Métricas de Desempenho no Estádio Olímpico Nilton Santos (Engenhão)

| Métrica | SCEN | CEN |
|---------------------|-------|-------|
| Total de Pacotes | 7200 | 7200 |
| Pacotes Perdidos | 2231 | 198 |
| Pacotes de Sucesso | 4969 | 7002 |
| Taxa de Sucesso (%) | 69.01 | 97.25 |
| Taxa de Falha (%) | 30.99 | 2.75 |

Esses dados destacam a vantagem da arquitetura CEN em ambientes desafiadores e de alta demanda, onde a rede enfrenta sobrecarga e interferências significativas. O Estádio Engenhão, sendo um local externo e sujeito a diversas condições de rede, representa um cenário complexo para a conectividade móvel. A arquitetura CEN demonstrou maior eficiência na entrega de pacotes, com uma taxa de sucesso de 97.25% e uma taxa de falha de apenas 2.75%. Em comparação, a arquitetura SCEN apresentou uma taxa de sucesso de 69.01% e uma taxa de falha de 30.99%. Esses resultados mostram como a CEN proporcionou menor perda de dados e melhor estabilidade de conexão.

6.3 Teste de Latência e Emissão de Alertas

Os testes de latência foram conduzidos no Estádio Olímpico Nilton Santos (Engenhão) e ao longo do trajeto de barca entre a cidade do Rio de Janeiro e Niterói, na Baía de Guanabara. O objetivo foi avaliar a latência do sistema de monitoramento de frequência cardíaca, configurado para enviar alertas ao detectar frequência cardíaca anômala continuamente por 5 minutos. Medimos o tempo necessário para processar e transmitir sinais do dispositivo de monitoramento para o servidor na nuvem, além do atraso na emissão de alertas em diferentes condições de rede (5G, 4G, 3G).

O sistema de monitoramento é configurado para detectar segmentos temporais nos quais a frequência cardíaca permanece elevada ou anormal por um período contínuo de 5 minutos. Essa duração foi escolhida com base em recomendações médicas e trabalhos relacionados, que sugerem que esse período é suficiente para indicar problemas cardíacos agudos ou crônicos que requerem atenção médica imediata.

Quando um segmento de 5 minutos com frequência cardíaca elevada é identificado, o sistema emite alertas para notificar os profissionais de saúde ou cuidadores responsáveis pelo acompanhamento do paciente. Ao analisar esses resultados, também examinamos a latência das arquiteturas CEN e SCEN, que impactam diretamente nos atrasos na emissão dos alertas.

Durante ambos os experimentos, foram observadas variações na qualidade da rede, resultando em intermitências entre 5G, 4G e 3G devido à alta demanda e sobrecarga da infraestrutura de rede e possíveis interferências. Essas condições reais permitiram medir a latência e o atraso na emissão de alertas em diferentes cenários de conectividade.

6.3.1 Resultados dos Testes de Latência. Os resultados mostrados na tabela a seguir destacam as diferenças de desempenho entre as arquiteturas:

Table 3: Teste de Latência - Cenário Estádio Olímpico Nilton Santos (Engenhão)

| Métrica | CEN | SCEN |
|---------------------------------|-----|------|
| Tempo de Resposta Médio 5G (ms) | 98 | 871 |
| Tempo de Resposta Médio 4G (ms) | 245 | 1314 |
| Tempo de Resposta Médio 3G (ms) | 383 | 2608 |

Table 4: Teste de Latência - Cenário Baía de Guanabara

| Métrica | CEN | SCEN |
|---------------------------------|------|------|
| Tempo de Resposta Médio 5G (ms) | 326 | 1951 |
| Tempo de Resposta Médio 4G (ms) | 754 | 2697 |
| Tempo de Resposta Médio 3G (ms) | 1229 | 4012 |

6.3.2 *Cálculo do Atraso Acumulado para Emissão de Alerta.* O atraso acumulado é uma medida do tempo total que se passa desde a detecção de um evento até a emissão de um alerta, considerando a latência na rede. Esse atraso é influenciado pela latência de envio de cada pacote de dados através da rede. A fórmula para calcular o atraso acumulado em minutos é dada por:

$$\text{Atraso Acumulado (min)} = \frac{\text{Tempo de Resposta Médio (ms)} \times 300}{1000 \times 60}$$

Os resultados dos testes de latência nos dois cenários, o Estádio Engenhão e a Baía de Guanabara, estão apresentados nas tabelas a seguir:

Table 5: Atraso Acumulado em Diferentes Condições de Rede no Estádio Olímpico Nilton Santos (Engenhão)

| Métrica | CEN | SCEN |
|---------------------------|------|-------|
| Atraso Acumulado 5G (min) | 0.49 | 4.36 |
| Atraso Acumulado 4G (min) | 1.23 | 6.57 |
| Atraso Acumulado 3G (min) | 1.92 | 13.04 |

Table 6: Atraso Acumulado em Diferentes Condições de Rede no Cenário Baía de Guanabara

| Métrica | CEN | SCEN |
|---------------------------|------|-------|
| Atraso Acumulado 5G (min) | 1.63 | 9.76 |
| Atraso Acumulado 4G (min) | 3.77 | 13.49 |
| Atraso Acumulado 3G (min) | 6.15 | 20.06 |

6.3.3 *Análise dos Resultados.* Os resultados dos testes indicam que a arquitetura CEN apresenta tempos de resposta significativamente menores que a arquitetura SCEN em ambos os cenários analisados. No Estádio Olímpico Nilton Santos (Engenhão), a arquitetura CEN obteve atrasos acumulados inferiores em todas as redes testadas (5G, 4G e 3G) em comparação com a SCEN. Já no cenário da Baía de

Guanabara, os atrasos foram maiores em relação aos experimentos realizados no estádio, possivelmente devido às condições do ambiente marítimo. O deslocamento da embarcação, a maior distância das torres de celular e a propagação do sinal sobre a água podem influenciar a qualidade da conexão, contribuindo para o aumento da latência observada ao longo do trajeto.

6.3.4 *Impacto do Atraso na Emissão de Alertas.* O atraso na emissão de alertas pode ter um impacto significativo na capacidade de resposta dos profissionais de saúde a situações críticas. No contexto do monitoramento contínuo de pacientes, especialmente em ambientes de grande porte e alta densidade como o Estádio, a rapidez na detecção e resposta a anomalias na frequência cardíaca pode ser crucial para salvar vidas.

- **Arquitetura CEN:** Com um atraso acumulado de aproximadamente 2 minutos no pior caso de rede identificado (3G), a arquitetura permite que os profissionais de saúde sejam notificados rapidamente sobre uma condição crítica. Essa agilidade pode ser vital para iniciar intervenções médicas imediatas e evitar complicações graves.
- **Arquitetura SCEN:** Em contraste, apresenta um atraso acumulado de aproximadamente entre 13 e 20 minutos no pior caso de rede identificado (3G). Esse tempo de resposta significativamente mais longo pode atrasar a notificação dos profissionais de saúde, reduzindo a janela de oportunidade para intervenções rápidas e potencialmente comprometendo a eficácia do atendimento médico.

7 CONSIDERAÇÕES FINAIS

A avaliação da arquitetura multicamadas para monitoramento remoto de saúde, baseada em computação em névoa e nuvem, comprovou sua eficácia em cenários de conectividade desafiadora. Os testes realizados em três ambientes distintos — o Estádio Olímpico Nilton Santos (Engenhão), o metrô do Rio de Janeiro, e a Baía de Guanabara — demonstraram que o processamento local nos *smartphones*, atuando como nós de névoa, foi fundamental para garantir a integridade dos dados e minimizar os impactos de instabilidades na rede. Essa abordagem se mostrou eficiente para o monitoramento em tempo real, oferecendo uma solução adaptável em condições variadas de conectividade.

Apesar de uma pequena perda de pacotes, possivelmente atribuída a interferências eletromagnéticas e à sobrecarga da rede, os resultados apontam para a necessidade de uma análise mais detalhada das condições de rede em cenários com alta mobilidade e congestionamento. O protocolo MQTT desempenhou um papel essencial na adaptação da arquitetura às variações de latência, assegurando a continuidade da coleta e transmissão de dados, mesmo diante de redes instáveis.

Com base nos resultados apresentados, a arquitetura CEN oferece baixa latência na entrega de notificações, otimizando o monitoramento dos pacientes e facilitando decisões rápidas e assertivas em cenários críticos, melhorando a gestão de eventos. Como trabalho futuro, os autores propõem investigar técnicas para reduzir interferências e sobrecargas em redes móveis, visando aumentar a confiabilidade e eficiência da solução em situações de monitoramento crítico.

8 AGRADECIMENTOS

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

REFERENCES

- [1] Zahra Ahmadi, Mostafa Haghi Kashani, Mohammad Nikravan, and Ebrahim Mahdipour. 2021. Fog-based healthcare systems: A systematic review. *Multimedia Tools and Applications* (2021), 1–40.
- [2] Manar Ahmed Kamal, Zainab Ismail, Islam Mohammad Shehata, Soumia Djirar, Norris C. Talbot, Shahab Ahmadzadeh, Sahar Shekoohi, Elyse M. Cornett, Charles J. Fox, and Alan D. Kaye. 2023. Telemedicine, E-Health, and Multi-Agent Systems for Chronic Pain Management. *Clinics and Practice* 13, 2 (2023), 470–482. <https://doi.org/10.3390/clinpract13020042>
- [3] Istabraq Al-Joboury and Emad Al-Hemiary. 2018. Performance Analysis of Internet of Things Protocols Based Fog/Cloud over High Traffic. *Journal of Fundamental and Applied Sciences* 10 (03 2018), 176–181. <https://doi.org/10.4314/jfas.v10i6s.113>
- [4] Aya Al-Sakran, Hazem Qattous, and Mohammad Hijjawi. 2018. A proposed performance evaluation of NoSQL databases in the field of IoT. In *2018 8th International Conference on Computer Science and Information Technology (CSIT)*. 32–37. <https://doi.org/10.1109/CSIT.2018.8486199>
- [5] Hamoud H. Alshammari. 2023. The internet of things healthcare monitoring system based on MQTT protocol. *Alexandria Engineering Journal* 69 (2023), 275–287. <https://doi.org/10.1016/j.aej.2023.01.065>
- [6] Nancy A Angel, Dakshanamoorthy Ravindran, P M Durai Raj Vincent, Kathiravan Srinivasan, and Yuh-Chung Hu. 2022. Recent Advances in Evolving Computing Paradigms: Cloud, Edge, and Fog Technologies. *Sensors* 22, 1 (2022). <https://doi.org/10.3390/s22010196>
- [7] H. Asri and Z. Jarir. 2023. Toward a smart health: big data analytics and IoT for real-time miscarriage prediction. *Journal of Big Data* 10, 1 (2023), 34. <https://doi.org/10.1186/s40537-023-00704-9> Received: 04 May 2022; Accepted: 23 February 2023; Published: 14 March 2023.
- [8] Sumit Bansal and Himanshu Aggarwal. 2022. Priority-based Cloud-Fog Architecture for Smart Healthcare Systems. (2022), 1–7.
- [9] Bilal Khalid Dar, Munam Ali Shah, Huniya Shahid, and Adnan Naseem. 2018. Fog Computing based Automated Accident Detection and Emergency Response System using Android Smartphone. In *2018 14th International Conference on Emerging Technologies (ICET)*. 1–6. <https://doi.org/10.1109/ICET.2018.8603557>
- [10] Mateus Gonçalo Do Nascimento, Gabriel Iorio, Thiago G Thomé, Alvaro AM Medeiros, Fabricio M Mendonça, Fernanda A Campos, José M David, Victor Ströele, and Mario AR Dantas. 2020. Covid-19: A digital transformation approach to a public primary healthcare environment. In *2020 IEEE Symposium on Computers and Communications (ISCC)*. IEEE, 1–6.
- [11] Bahar Farahani, Farshad Firouzi, Victor Chang, Mustafa Badaroglu, Nicholas Constant, and Kunal Mankodiya. 2018. Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare. *Future Generation Computer Systems* 78 (2018), 659–676.
- [12] Niroshinie Fernando, Seng W. Loke, and Wenny Rahayu. 2013. Mobile cloud computing: A survey. *Future Generation Computer Systems* 29, 1 (2013), 84–106. <https://doi.org/10.1016/j.future.2012.05.023> Including Special section: AIRCC-NetCoM 2009 and Special section: Clouds and Service-Oriented Architectures.
- [13] Eliza Gomes, Rubens Zanatta, Patricia Plentz, Carlos De Rolt, and Mario Dantas. 2020. An Approach of Time Constraint of Data Intensive Scalable in e-Health Environment. In *International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*. Springer, 158–169.
- [14] Bhole Rahul Hiran, Chapte Viresh M., and Karve Abhijeet C. 2018. A Study of Apache Kafka in Big Data Stream Processing. In *2018 International Conference on Information, Communication, Engineering and Technology (ICICET)*. 1–3. <https://doi.org/10.1109/ICICET.2018.8533771>
- [15] Abeer Ilyas, Mohammed Alatawi, Yasir Hamid, Saeed Mahfooz, Islam Zada, Nee-lam Gohar, and Mohd Asif Shah. 2022. Software architecture for pervasive critical health monitoring system using fog computing. *Journal of Cloud Computing* 2022 (11 2022), 84. <https://doi.org/10.1186/s13677-022-00371-w>
- [16] Mostafa Haghi Kashani, Mona Madanipour, Mohammad Nikravan, Parvaneh Asghari, and Ebrahim Mahdipour. 2021. A systematic review of IoT in healthcare: Applications, techniques, and trends. *Journal of Network and Computer Applications* 192 (2021), 103164.
- [17] James Manyika, Michael Chui, Peter Bisson, Jonathan Woetzel, Richard Dobbs, Jacques Bughin, and Dan Aharon. 2015. Unlocking the Potential of the Internet of Things. *McKinsey Global Institute* 1 (2015).
- [18] Fabricio Martins Mendonça, Mário A. R. Dantas, Wallace T. Fortunato, Juan F. S. Oliveira, Breno C. Souza, and Marcelo Q. Filgueiras. 2022. Wearable Devices in Healthcare: Challenges, Current Trends and a Proposition of Affordable Low Cost and Scalable Computational Environment of Internet of Things. In *XXVII Brazilian Congress on Biomedical Engineering*, Teodiano Freire Bastos-Filho, Eliete Maria de Oliveira Caldeira, and Anselmo Frizzera-Neto (Eds.). Springer International Publishing, Cham, 1301–1308.
- [19] Marcelo Q. Filgueiras Juan F. S. Oliveira Paulo C. S. Vidal, Ronaldo M. Salles. 2024. Desenvolvimento e avaliação de uma Arquitetura para monitoramento remoto em saúde utilizando fog e cloud computing. *Artigo aceito no XX Congresso Brasileiro de Informática em Saúde (CBIS)* (2024).
- [20] Yuya Sasaki and Tetsuya Yokotani. 2019. Performance Evaluation of MQTT as a Communication Protocol for IoT and Prototyping. *Advances in Technology Innovation* 4, 1 (Jan. 2019), 21–29. <https://ojs.imeti.org/index.php/AITI/article/view/2522>
- [21] Shereena Shaji, Ravi Sankaran, Ramesh Guntha, and Rahul Krishnan Pathinarupothi. 2023. A Real-time IoMT Enabled Remote Cardiac Rehabilitation Framework. In *2023 15th International Conference on COMMUNICATION SYSTEMS NETWORKS (COMSNETS)*. 153–158. <https://doi.org/10.1109/COMSNETS56262.2023.10041272>
- [22] Pankaj Sharma and P K Gupta. 2023. Optimization of IoT-Fog Network Path and fault Tolerance in Fog Computing based Environment. *Procedia Computer Science* 218 (2023), 2494–2503. <https://doi.org/10.1016/j.procs.2023.01.224> International Conference on Machine Learning and Data Engineering.
- [23] Georgios L Stavrinides and Helen D Karatza. 2019. A hybrid approach to scheduling real-time IoT workflows in fog and cloud environments. *Multimedia Tools and Applications* 78, 17 (2019), 24639–24655.
- [24] Hubert Tardieu, David Daly, José Esteban-Lauzán, John Hall, and George Miller. 2020. Case study 2: the digital transformation of health care. In *Deliberately Digital*. Springer, 237–244.
- [25] A Tina Victoria and M. Kowsigan. 2022. Secure Management of Healthcare Data in Fog and IoT Networks: A Short Survey on Existing Security Protocols. In *2022 3rd International Conference on Smart Electronics and Communication (ICOSEC)*. 512–518. <https://doi.org/10.1109/ICOSEC54921.2022.9952038>
- [26] Pedro H Vilela, Joel JPC Rodrigues, Luciano R Vilela, Mukhtar ME Mahmoud, and Petar Solic. 2018. A critical analysis of healthcare applications over fog computing infrastructures. In *2018 3rd International Conference on Smart and Sustainable Technologies (SpliTech)*. IEEE, 1–5.