

Adaptative vital signs monitoring system based on the early warning score approach in smart hospital context

Imen Ben Ida¹ | Moez Balti^{1,2} | Sondes Chaabane³ | Abderrazak Jemai^{1,4}

¹Polytechnic School of Tunisia, Electronic Systems and Communications Networks Laboratory (SERCOM), Carthage University, Tunis, Tunisia

²IsetCom, Carthage University, Tunis, Tunisia

³Hauts-de-France Polytechnic University, CNRS UMR 8201 – LAMIH – Laboratory of Automatic Mechanics and Industrial and Human Informatics, F-59313 Valenciennes, France

⁴INSAT, Carthage University, Tunis, Tunisia

Correspondence

Imen Ben Ida, Electronic Systems and Communications Networks Laboratory (SERCOM), Polytechnic School of Tunisia, Carthage University, Tunis, Tunisia.

Email: Imen.benida@gmail.com

Abstract

Changes in vital signs are an important indicator of physiological decline and provide opportunities for early recognition and intervention. The collected vital signs data can be evaluated using several approaches such as the Early warning score (EWS) approach to predict the risk level of patients. By exploring the Internet of things (IoT), vital signs monitoring solutions are automated based on various medical devices and sensors. However, there is a lack of efficient tools that enable an adaptative monitoring depending on the patient situations. This article explores the IoT technologies to provide an EWS system in smart hospital situation. The proposed solution presents an adaptative configuration of the vital signs monitoring process depending on the patient's health status variation and the medical staff decisions. Also, an intelligent notification mechanism that reduces the delay of the medical staff intervention in the case of risk detection is proposed.

1 | INTRODUCTION

Since the beginning of the 20th century, patient surveillance has been performed in hospitals by measuring the same vital signs. These typically consist of blood pressure, heart rate, temperature, and respiratory rate, with the recent addition of oxygen saturation. The guidelines from the National Institute for Health and Clinical Excellence (NICE) recommend the observation of six vital signs as a minimum, including oxygen saturation in addition to the five others used in the original score [1]. Large amounts of vital signs data are routinely collected in hospitals in order to better diagnose patient health status and to identify possible anticipatory actions.

Several medical approaches are used to evaluate the collected data. A prevalent example is the Early warning score (EWS) approach which has been in use for several years as a tool to predict the risk level of patients. It was proposed for the first time as a paper-based method needing periodical check-ups to assign a score based on patient's vital signs (i.e. heart rate, respiration rate, body temperature, blood pressure). The score of each medical sign depends on the non-respect of a predefined normal interval. The summation of all scores reflects the global patients risk level [2]. Since 85% of severe adverse events (SAE) are

preceded by abnormal vital signs, the vital signs monitoring based on EWS approach have evolved as a means of alerting health professionals to patient clinical decline [3].

By exploring the Internet of things (IoT) technologies, vital signs control is automated based on various medical devices and sensors. The use of automated systems reduces the errors of the manual EWS systems [4] and facilitates the nurses' functions such as constantly gathering and storing the vital signs records.

The emergence of the IoT, the electronic records and the computerized transaction systems have improved the efficiency and effectiveness of EWS systems. As part of the IoT, medical devices such as radio-frequency identification (RFID) tags and wearable sensors have been significantly expanded in scope [5]. This proliferation of devices is expected to revolutionize health care by speeding up treatment and diagnostic processes, monitoring of health signs for 24/7, and enhancing patient care quality.

By exploring the IoT architecture, there are currently two important challenges of EWSs systems that need to be considered. The first challenge is how to ensure a personalized monitoring of patients' vital signs depending on their situations and the medical experts' requirements especially in the case of controlling an important number of patients. The second

challenge is the need of timely response of medical staff in case of problem detection.

Our work is motivated by the challenges described above, and its main objectives are

- The development of an EWS system for a smart hospital environment.
- The use of a self-adaptative configuration algorithm to change the control frequency of the vital signs data depending on the patient situations.
- The use of an effective notification mechanism that takes in consideration the medical staff activities.

This article is organized as follows: In Section 2, some related works are highlighted. In Section 3, we describe the vital signs evaluation with EWS systems. We present the vital signs monitoring in smart hospital context in Section 4. Our proposed solution is detailed in Section 5 and summarized in Section 6. The implementation and the evaluation are presented and discussed in Section 7. Section 8 presents the concluding remarks and future work.

2 | RELATED WORKS

In [6], authors present a solution which takes benefits from the concept of Edge computing in the context of IoT-based EWSs systems. The solution provides high level services in a Geo-distributed fashion at the edge of the network. The proof of concept is demonstrated with smart e-Health Gateway called UT-GATE implemented for an IoT-based remote health monitoring system. The demonstration includes the data flow processes from data acquisition at sensor nodes to the cloud and the end-users. However, the data has a static interval of acquisition which is defined at the development stage. As a result, the solution does not give the possibility of personalization. The smart e-Health Gateway proposed in [7] provides local storage to perform real-time local data processing and mining. When a patient's vital signal is processed, reliable IoT systems are provided to facilitate fault-tolerant healthcare services. Zhang et al proposes in [8] a patient-centric cyber-physical system named Health-CPS aiming to ensure convenient Healthcare service. The Health-CPS depends on Cloud computing and data analytics to handle the big data related issues of different healthcare applications. The system is composed of several layers such as data collection layer, data management layer and data-oriented service layer. The system collects data in a unified standard. It supports distributed storage and parallel processing.

In [9], authors presented a self-aware EWS system with a fuzzified reliability validation which recognizes erroneous vital signs caused by various measurement artefacts such as loose sensors, detached sensors or other interferences. The proposed system was successful in detecting such events and decreased the data reliability during such events.

In [10] the authors propose a Smart Hospital System (SHS) which relies on different technologies, specifically RFID, WSN,

and smart mobile, interoperating with each other through a CoAP/6LoWPAN/REST network infra-structure. The SHS is able to collect, in real time, both environmental conditions and patients' vital signs via an ultra-low-power Hybrid Sensing Network (HSN) composed of 6LoWPAN nodes integrating UHF RFID functionalities. Sensed data are delivered to a control centre where an advanced monitoring application makes them accessible by both local and remote users via a REST web service.

The authors present in [11] a low-cost health monitoring system that provides continuous remote monitoring of the ECG and automatic analysis and notification. The system consists of energy-efficient sensor nodes and an edge layer that take full advantage of IoT. The sensor nodes collect and wirelessly transmit ECG, respiration rate, and body temperature to a smart gateway which can be accessed by appropriate care-givers. In addition, the system can represent the collected data in useful ways.

A gateway and Semantic Web enabled IoT architecture is proposed in [12] to provide interoperability between different systems, using established communication and data standards. The proposed Semantic gateway as Service (SGS) translates the messages from different protocols such as XMPP, CoAP and MQTT via a multi-protocol proxy architecture. Authors did not mention the characteristics of the used hardware and they do not include mechanisms for flexible applications' allocation, which would consequently bring benefits in terms of energy efficiency.

In [4] a context-aware EWS system using an IoT-based solution is proposed. It is implemented for in-home patients with the possibility of monitoring the activity level of patients and environmental properties to ensure that patients are given care in a standard situation and the EWS scoring data is captured in a suitable condition. The scores are calculated at the cloud layer which could results a problem of latency special in emergency cases. Added to that, the system supports only the change of score calculation frequency of EWS calculation depends on the patient's medical history and earlier scores. It also supports manual measurements such as blood pressure and blood glucose and sends notification to patients to enter the manually measured values to the system.

All the previous proposed solutions require different technology skills to modify or to scale out an existing system in a hospital context. The devices list is predefined at the conception level with fixed parameters of configuration. Added to that, the frequency of collecting and saving data are defined by the developers at the level of implementation and do not support the possibility of modification after deployment in a real case. As a result of this static configuration, a device collect data with the same parameters defined from the first step of development and it cannot be reused for personalized cases.

However, vital signs data are different from other collected data in IoT environments as it depends on the no stable situation of the patient's health. For example, the frequency of the collected respiration data with a sensor from a patient depends on the respirator illness danger. In some cases, an

interval of 1 h is sufficient but in other cases 1 min is needed. If the storage operation of the data is unique, an important unnecessary information will take place in the memory and demand useless process. The healthcare systems in hospitals should make it easy to personalize their parameters to a particular patient and health condition. This will lead to more accurate responses. Added to that, there is a need to an intelligent notification mechanism to ensure a timely intervention of the medical staff in the case of abnormal values detection.

3 | VITAL SIGNS EVALUATION WITH EARLY WARNING SCORE SYSTEMS

In hospitals, particularly in intensive care units, the EWS is a prevalent tool, by which the patient's vital signs are periodically recorded and the emergency level is interpreted [4]. To this end, a score (0 for a perfect condition and 3 for the worst condition) is allocated to each vital sign according to its value and the predefined limits (see Figure 1). The summation of the obtained scores indicates the degree of health deterioration of the patient (the higher the EWS, the worse the patient's health condition).

3.1 | National early warning score (NEWS)2 approach

NEWS2 chart illustrated in Figure 1 is a revised version of NEWS chart. The NEWS was created to standardize the process of recording, scoring and responding to changes in routinely measured physiological parameters in acutely ill patients. It was developed to improve the response to clinical deterioration in patients with acute illness. In [13] NEWS was evaluated against a range of outcomes that are of major importance to patients and staff. It demonstrates a good ability

to discriminate patients at risk of the combined outcome of cardiac arrest, unanticipated Intensive Care Unit ICU admission or death within 24 h, which provides ample opportunity for an appropriate clinical intervention to change patient outcome [14].

NEWS2 could be made safer for patients with hypercapnic respiratory failure by having two scoring systems for (saturation pulse oxygen) SpO2:

- The existing SpO2 scoring system (Scale 1) that would apply to the majority of patients.
- A dedicated SpO2 scoring system for patients with hypercapnia respiratory failure (Scale 2) whose desired oxygen saturations are set at a lower level (88%–92%), with the NEWS scoring system adjusted accordingly.

3.2 | Early warning score systems requirements

Added to the basic functionality of score calculation, an efficient EWS system should take in consideration several requirements such as:

- Personalization support:
The patient's vital signs data are the key to successful medical decisions. Each patient requires a personalized control depending on his situation [15]. For example, when a patient has a disease that causes hyperthermia, the temperature is a critical data which must be visualized with more precision every second. While other non-critical parameters can be calculated only every hour.
- Medical staff engagement
The EWSs systems are highly user-dependent and depend on the appropriate response and actions of the Medical Emergency Teams [3].

Physiological parameter	Score						
	3	2	1	0	1	2	3
Respiration rate (per minute)	≤8		9–11	12–20		21–24	≥25
SpO ₂ Scale 1 (%)	≤91	92–93	94–95	≥96			
SpO ₂ Scale 2 (%)	≤83	84–85	86–87	88–92 ≥93 on air	93–94 on oxygen	95–96 on oxygen	≥97 on oxygen
Air or oxygen?		Oxygen		Air			
Systolic blood pressure (mmHg)	≤90	91–100	101–110	111–219			≥220
Pulse (per minute)	≤40		41–50	51–90	91–110	111–130	≥131
Consciousness				Alert			CVPU
Temperature (°C)	≤35.0		35.1–36.0	36.1–38.0	38.1–39.0	≥39.1	

FIGURE 1 The NEWS2 scores chart

- Need for expert's decision

The warning scores cannot replace the specialist's decision and the importance on knowing individual patients, and they cannot also replace the background to the observations that are recorded. NEWS correlated poorly with the patient's clinical status within the first 24 h post-operatively [2].

4 | VITAL SIGNS MONITORING IN SMART HOSPITAL CONTEXT

Patient vital signs are important indicators of the health status. With the advent of the IoT, several interconnected objects can be used to improve the collection and the process of vital signs with partially or fully automatized methods. The use of the Edge computing technologies in the IoT environment has the potential to reduce costs, increase quality of life, and improve the quality of medical services. In the following sub-paragraphs, we present the contextual architecture and the different services that can offer the Edge layer.

4.1 | Smart hospital architecture

Figure 2 depicts the contextual architecture considered in our solution which is a smart hospital environment with intelligent interconnected devices. The majority of these devices, for example wearable medical sensors, are not capable of storing data they generate. A straightforward design approach is to transfer this data to a cloud for processing. However, the latency of the connection with the cloud could be significant because of the large number and the resources limits of the connected devices. Edge Computing [15] is an essential paradigm shift towards a hierarchical system architecture and a more responsive design. As shown in Figure 2, Edge is an intermediate computing layer between the cloud and end devices that complements the advantages of cloud computing by providing additional services for the emerging requirements in

the field of IoT. It is exploited by reinforcing edges such as gateways with sufficient processing power, intelligence, and orchestrated networking capabilities to provide efficient healthcare services without limiting the mobility of patients.

Edge-based system can be organized in a distributed manner across the three layers, the perception layer, the edge layer and the cloud layer.

4.1.1 | The perception layer

This layer is the lowest layer that has contact with the monitored patients. Various sensors collect medical data including the physiological sign data like heart rate and temperature, environmental data and other data like nurse's presence data. They typically have communication capabilities in one of the many possible technologies in order to ensure the acquisition of different types of data.

4.1.2 | The cloud layer

The cloud layer can be obtained either via Internet connected remote servers or by local servers connected to local hospital information system (HIS) to provide more services such as data analytics and security [16].

The system could also enable healthcare staff to receive instant notifications, feedback and setting adjustments via an administration control panel.

The integration of Cloud Computing with IoT technologies for patient data monitoring, provides several capabilities such as storage, processing, scalability and networking. It is able to simplify the healthcare processes and allows to enhance the quality of the medical services [17].

4.1.3 | The edge layer

The edge layer reinforces the devices layer with storage, processing and communication capabilities in order to fulfil different requirements in a healthcare system. Several functionalities can be embedded in smart gateways in order to distribute the data processing and to ensure real-time responsiveness. A gateway at the edge layer acts as a dynamic touching point between a sensor network and the local/cloud server. It receives data from different sub-networks and offers protocol conversion and other services such as data aggregation and signal filtering. The collected data are stored in local databases before being sent to the hospital server and used to evaluate a patient's risk of health deterioration based on different strategies such as EWS.

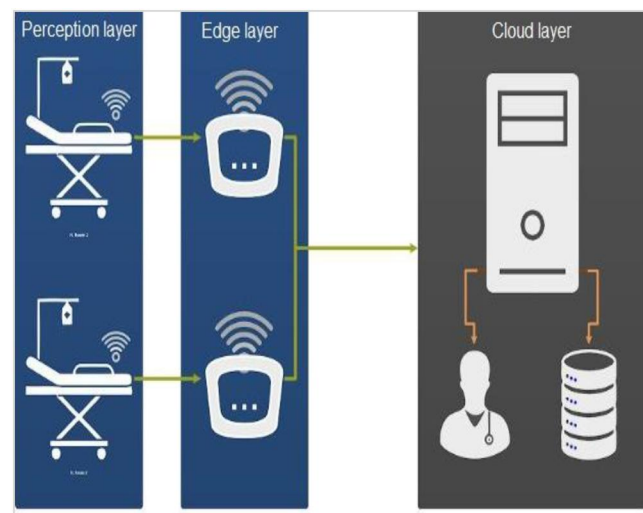


FIGURE 2 Smart hospital architecture

4.2 | Edge-based systems requirements

Recent advancements in IoT and specially the use of the Edge computing technologies can mitigate the restrictions of manual EWS manual approach by providing continuous remote health

monitoring. In EWS systems based on IoT devices, patients' vital signs are continuously monitored via mobile/wearable sensors, while server performs data analysis and decision-making algorithms for the score determination. However, there still other requirements that should be taken in consideration when developing an Edge-based EWS system.

- **Interoperability**

In a smart hospital, heterogeneous components must be able to exchange data and services. Interoperability can be viewed from network, syntactic, and semantic perspective.

A network interoperability allows the exchange of patient's data across different networks, potentially using different communication technologies. Syntactic interoperability is the formatting and encoding of any exchanged information or service. Semantic interoperability indicates information or a service. It allows the interchange between the ever growing and changing set of devices and services in IoT-based system [18].

- **Quality of services**

The vital signs monitoring system performance ultimately helps in achieving global healthcare system performance. It should remain operational for the duration of a patient stay, even in the presence of failures. Every component or service needs to be reliable to achieve an overall performance, which includes communication, data, technologies, and devices from all layers. For example, transferring continuously the huge amount of data generated by IoT medical devices requires high bandwidth. As a result, there is a requirement of obtaining adequate network performance to transfer vital signs data without interruption [19].

- **Energy contrast support**

The continuous connectivity to send vital signs data results high consumption of energy. IoT devices have limited processing and energy capabilities that do not allow complex, data processing and do not permit to continuously send real time data to the server efficiently. Added to that, changing the energy battery frequently is not possible for wearable and planted shields [19].

- **Scalability**

A vital sign monitoring system needs to be scalable to accommodate growth in the IoT's network and applications/services in smart hospital. Considering the size of the IoT's network, loose coupling and/or virtualization mechanisms are useful in improving scalability, especially application and service level scalability, by hiding the complexity of the underlying hardware or service logic and implementation [18].

- **Real time or timeliness**

Delayed information or services in emergency cases can make the vital signs monitoring system useless and even dangerous.

- **Security and privacy**

An important requirement is how to provide appropriate authorization rules and policies while ensuring that only

permitted users have access to the sensitive data and only authorized devices are sending correct information. Moreover, there is a challenge of data privacy as public key cryptography cloud is not applied to all layers because of the processing power constraints imposed by IoT objects [20].

5 | THE PROPOSED SOLUTION

Our proposed solution supports a patient-centric process by providing the ability to tailor vital signs monitoring parameters to a specific health condition or treatment. The solution is based on the use of a message-oriented middleware. The following paragraphs describe the solution architecture and the supported communication model.

5.1 | Publish-subscribe communication model

Our solution uses the publish/subscribe pattern for the data exchange between the different architecture layers.

Publish-subscribe messaging systems support data-centric communication and have been widely used in IoT systems. With the publish-subscribe pattern, the exchange of messages between clients is ensured using a broker that manages topics and sub-topics. A publisher on a given topic can send messages to other clients acting as subscribers to the topic without the need to know about the existence of the receiving clients [15].

To organize the topics and sub-topics in both local and cloud brokers, we propose a common model presented in Figure 3.

We define four categories of operations:

- **Perception:** It means the collection of time-series data sensed by the medical devices. These devices are small constraint-resources elements that generate the measured data with predefined fixed interval of time. The devices publish the collected data as messages to the broker at the gateway layer.
- **EWS score:** The corresponding score of each sensed data depending on the NEWS2 specification is illustrated in Figure 1.

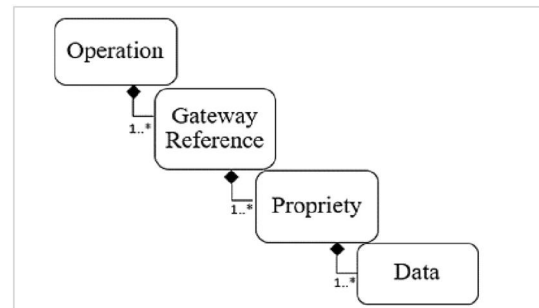


FIGURE 3 The communication model

- **Configuration:** The definition of the gateway parameters by the medical staff. For example, the frequency of saving the collected data.
- **Notification:** They are the alert messages that concern the devices status and the vital signs scores.

For each category, there are corresponding proprieties. First, the proprieties of the perception are the names of the collected data such as temperature and heart beats. Second, the proprieties of configuration are the status of each concerned data and the time range for both local and cloud storage. For example, to disable the collection of temperature in the gateway1, the cloud broker sends the following message:

Configuration/gateway1/temperature/False

'False' is the message value that will be received by the client subscriber of the topic:

'Configuration/gateway1/temperature'

On the other side, to publish a body temperature value equal to 37° which is collected by the gateway1, the message is: Perception/gateway1/temperature/37

For the EWS score, the propriety can be the name of the corresponding vital sign or the global score. The proprieties of the notification operation are the references of the concerning person that must be notified in case of device dysfunction or abnormal score detection. For example, a nurse who has a reference code "N102" is subscribed to.

'Notification/gateway1/N102/scorealert'.

5.2 | Edge layer components

We propose the use of a gateway as an edge layer to put the computing at the proximity of data sources. In each hospital room, a smart gateway is used to collect the vital signs of the patients. The main functionalities of the gateway are as follows:

- Collecting the vital signs of the patient staying in the corresponding room.
- Health status evaluation of each patient by calculating the EWS.
- Ensuring the medical staff intervention in case of the detection of high level EWS.
- Sending notifications in case of a device connexion failure or dysfunction.

To ensure the interventions after sending notifications, we include the staff activities which are frequently adapted in hospital context. The main components of the proposed gateway illustrated in Figure 4 are as follows:

5.2.1 | Local broker

The Broker server acts as an intermediary for messages sent between the publishers and the subscribers for a specific topic. It routes the messages based on topic rather than the IP address. When a message is sent by the publish client to the

broker server, all the subscribe clients interested on the related topic of the message will receive the publication. Medical devices are considered as data publishers. A topic is the device name and for each device, a subscriber is created to listen to the published data and save it in a local database with the required parameters. The main component of the gateway is an embedded configuration service that provides medical staff with the ability to log into and configure the data process behaviour depending on patients' situations. The Medical staff can configure publishers and subscribers by providing information such as the ID of the devices, the references of the patients and the corresponding dates of hospitalization.

5.2.2 | The virtual semantic sensor

It is a software component with an objective to translate the sensed data into higher semantic information. The input of a virtual sensor is a physical stimulus sent by a sensor. The sensor can be a vital signs monitoring device of RFID lecture that captures the existence of the medical staff. The output of virtual semantic sensor is a new personalized description of semantic sensor observations.

A virtual sensor has three main roles. First it saves the received data as a significant information in the local database. Second, it calculates the score of the data depending on its type and using the predefined intervals of the NEWS2 approach. In the case of problem detection with the calculated score or data interruption, the virtual sensor publishes a notification to the local broker. Added to that, each virtual sensor is subscribed to a configuration topic in order to receive the requested interval of saving data in the local database. To ensure the privacy of the patient information, the virtual sensor does not store the patients' details such as their names or their ages. It stores only the references of patients associated with the devices' numbers and the corresponding range time of data collection.

5.2.3 | Scores manager

The scores manager is a software component that receives calculated scores from all the virtual sensors. It calculates the global score of each patient and evaluates its risk level. The calculated scores are saved in the local database and they are sent regularly to the hospital server for long term storage. In case of high score detection, the scores manager keep publishing significant notifications to the local broker until it receives a validation of the medical staff intervention. The notification management approach is detailed in Section 5.3.

5.2.4 | Configuration manager

It is in an intermediate position between the hospital server and the virtual semantic sensors. It initializes each new virtual sensor with the corresponding configuration. The configuration manager of each gateway is considered as a subscriber to

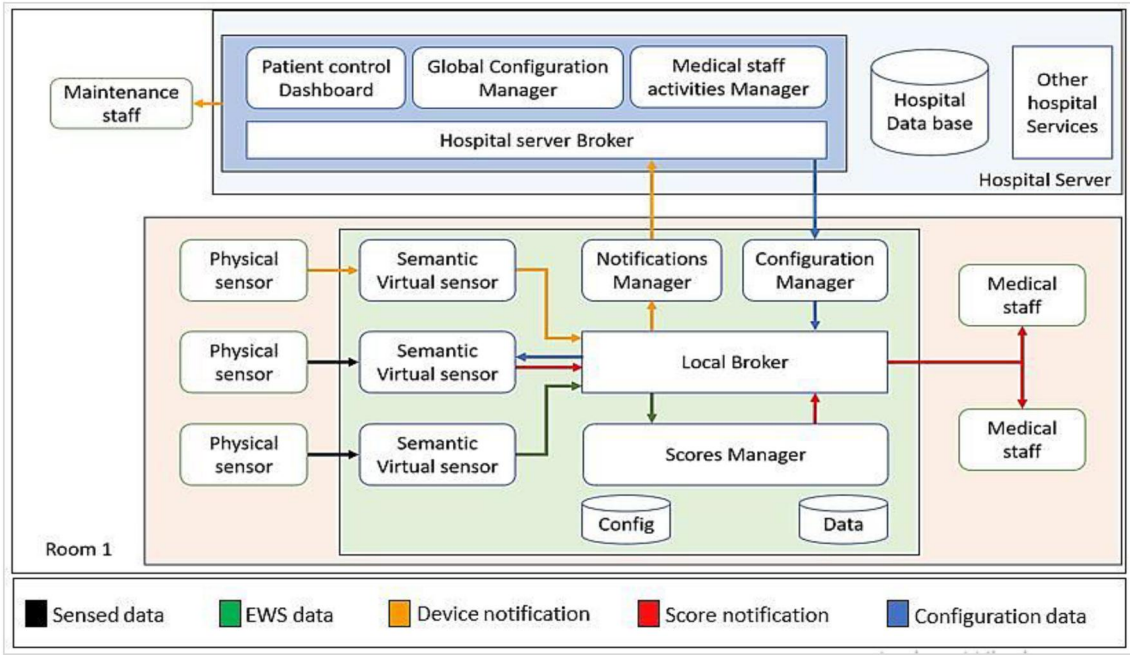


FIGURE 4 The solution architecture

the corresponding configuration topic defined at the hospital server. Two types of configuration are supported by the configuration manager.

- Manual configuration

The medical staff can submit a configuration request using a web interface, the server broker publish the received request as a configuration message to the concerned gateway. If the request is about adding a new device, a new virtual sensor is created after validation. In case that the medical staff choose to modify some parameters of an existing device, the corresponding virtual sensor checks the different inserted parameters: device id, patient Ref and the time range of saving data and saves the new configuration. The separation of the virtual sensors as independent subscribers ensures that the modification or the elimination of a device does not affect the whole system. The principal configuration that we consider in our work is the change of the frequency of data collection and storage depending on the patient situation requirements. The care givers change this frequency using a web application at the server level

- Self-adaptive configuration

Added to the manual configuration possibility, our system integrates an automation algorithm illustrated in Algorithm one for the calibration of the data storage and the score calculation frequency.

For each score level, the maximum value is defined as score level limit. The configuration manager reads frequently the calculated score of each vital sign from the local database and compares it with the corresponding predefined limits.

If it captures successive abnormal values, the frequency of the collection of this data is automatically increased to be able

to monitor the data concerned with more precision. The new required frequency value is saved in the local database and published as a message to the corresponding virtual semantic sensor. When the data becomes more stable and less than the predefined limits, the administrator can decide to change this frequency. The actual version of our system supports three values of data saving frequencies which are 1 h, 1 min and 1 s. These values can be changed depending on the medical experts' requirements.

Algorithm 1 Auto-adaptive configuration.

Inputs: (SC) NEWS2 calculated score, (Freq) Corresponding frequency, (VST) Virtual Sensor Topic.

Variables: (Score0Limit, Score1Limit) NEWS2 scores limits.

Output: NewFreq (new data collection frequency).

```

00: begin
01: If ((Freq = FreqLevel1) and (SC = 0)) then
02: Score0Limit = GetScoreLimit (VST, 0)
03: If (verifOf2LastValues (VST, SC, Score0Limit) = true) then
04: NewFreq = FreqLevel2
05: End if
06: Else if ((Freq = FreqLevel2) and ((SC = 1) or (SC = 2))) then
07: Score1Limit = GetScoreLimit (VST, 1)
08: If (verifOf3LastValues(VST, SC, Score1Limit) = true) then
09: NewFreq = FreqLevel3
10: End if
11: End if
12: Publish (VST, NewFreq)
13: End

```

According to the last saved SC score level and the most recent Freq data saving frequency, the program retrieves the limit values corresponding to this score level according to the NEWS2 approach. Then, if the last score is 0, the `verifOf2LastValues` function detailed in Algorithm two compares the difference between the last two saved `DiffValues` and the difference between the score level threshold value and the last saved `DiffScoreValue`. If the `DiffScoreValue` difference is less than or equal to the value of `DiffValues`, the interval of saving the data `NewFreq` is reduced. If the last calculated score is greater than 0, the program calls the `verifOf3LastValues` function detailed in algorithm 3.

Algorithm 2 *Data comparison in the case of score 0.*

Inputs: (SC) NEWS2 calculated score, (ScoreOLimit) limit value of the first score class (score 0), (VST) Virtual Sensor

Topic.

Variables: (vt, vt - 1) last two device's values. (DiffValues, DiffScoreValue) calculated differences

```

00: begin.
01: Read (vt, vt - 1)
02: DiffValues = vt - vt - 1
03: DiffScoreValue = ScoreOLimit-vt
04: If (DiffScoreValue ≤ DiffValues) then
05: Return TRUE
06: Else return FALSE
07: End if
08: End

```

The `verifOf 3LastValues` function detailed in Algorithm 3 has the same principle of the `verifOf 2LastValues` function but it compares the `DiffScoreValue` value to the last two calculated differences. The current version of our system supports three data saving interval values which are 1 h, 1 min, and 1 s. These values can be changed according to the needs of medical experts.

Algorithm 3 *Comparison of data in the case of score 1 or 2.*

Inputs: (SC) NEWS2 calculated score, (ScoreLimit) limit value of the score class, (VST) Virtual Sensor

Variables: (vt, vt - 1, vt - 2) last device's values. (DiffValues1, DiffValues2, DiffScoreValue) calculated differences

```

00: begin
01: Read (vt, vt - 1, vt - 2)
02: DiffValues1 = vt - vt - 1
03: DiffValues2 = vt - 1 - vt - 2
04: DiffScoreValue = ScoreLimit - vt
05: If ((DiffScoreValue ≤ DiffValues1) or
(DiffScoreValue ≤ DiffValues2)) then
06: Return TRUE
07: Else return FALSE
08: End if
09: End

```

5.3 | Notification management approach

Ensuring the right notification at the necessary time is a primary requirement for a risk evaluation system. However, there is a need of efficient physical intervention after receiving a risk detection notification. To ensure a quick intervention, our solution takes in consideration the medical staff activities that are updated frequently depending on the patients' and the hospital needs. A software component which we called Medical staff activities manager in Figure 4 is considered as a subscriber to all the activity topics. Each medical staff updates his activity status using a mobile application and the RFID badge. For example, when a nurse changes her place to control a patient in a hospital room, she passes her badge in front the RFID lecture of the corresponding room.

A default planning of all medical staff is defined every day by the hospital administration. However, this planning is flexible and can be updated. A default number of nurses is affected for each room which contains vital signs monitoring devices in the hospital. The medical staff activities manager sends frequently the corresponding nurses list to each gateway in the rooms. In case of score notification, the local notifications manager publishes a notification message to this list and wait for a validation message from one of the corresponding nurses. If it does not receive any message after a predefined period of time, it publishes the request to the hospital server broker and the global activities manager looks for other responsible nurses to inform them. The notification process continue until a message from the RFID lecture in the concerned room is received and ensures that a nurse is coming to control the patient with a high score value.

Added to that, the solution controls the validity of the sensed data by comparing each received value to a predefined allowed interval that depends on the data type. For example, the accepted data range of the body temperature is [36 ... 41]. If the device sends a wrong data, a notification is sent by the gateway to the hospital server broker. In Figure 5 we illustrate the notification process.

6 | SUMMARY OF CONTRIBUTIONS

The main contribution of this article is the implementation of a self-adapting early warning system for vital signs control in the context of a smart hospital. To ensure a personalized monitoring that takes into account the variation in health status, the proposed system supports two types of configurations, a self-adaptive configuration and a manual configuration, which depend on the decisions of the medical staff. In particular, the system provides the ability to change the storage interval of vital signs data at the edge layer in order to customize device behaviour. We also provide an effective notification approach by considering the activities of medical staff which are frequently updated based on patients' and hospital needs.

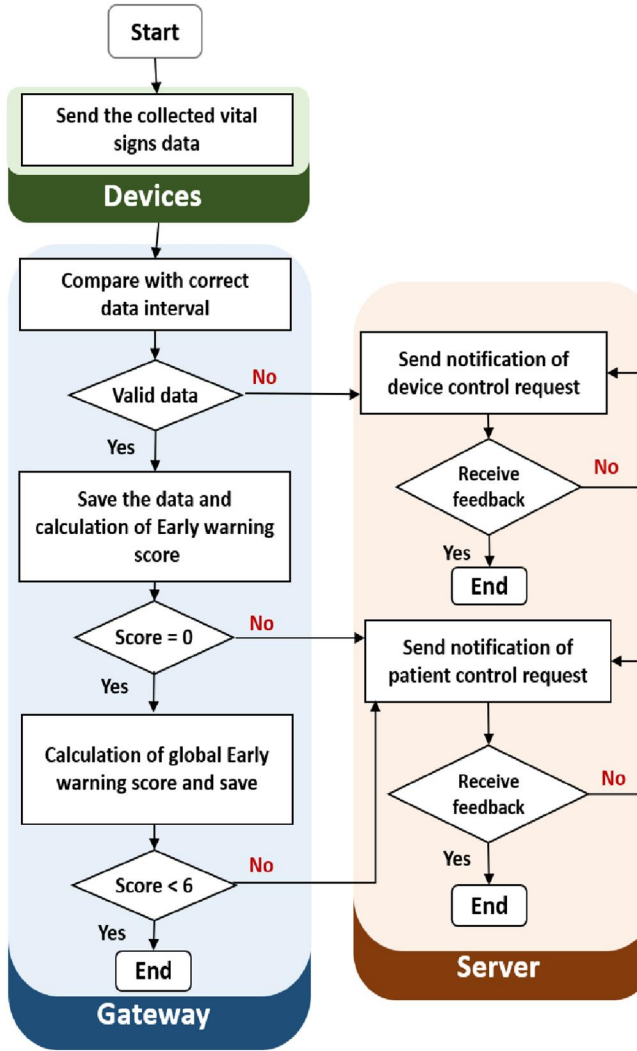


FIGURE 5 The notification process flowchart

7 | IMPLEMENTATION AND EVALUATION

7.1 | Implementation

The Gateway is a Raspberry Pi three which is a low-cost small-size single board with 1 GB of RAM and 1.2 GHz processor [21]. As an Operating System for the gateway, we use Raspbian the pre-compiled Debian OS which is especially optimized for the Raspberry Pi.

We install the InfluxDB database which is an open-source Time Series database written in Go. At its core is a custom-built storage engine called the Time-Structured Merge (TSM) Tree, which is optimized for time-series data. InfluxDB supports SQL-like query languages named InfluxQL and has the advantage of easy scale-out. It provides support for mathematical and statistical functions across time ranges; also it is developed for custom monitoring, metrics collection and real-time analytics [22].

We use the mosquito implementation as a messages' broker and Node JS clients as subscribers. [23].

Node JS is based on Google's runtime implementation named 'V8' engine [24]. V8 and Node are mostly implemented in C and C++, focusing on performance and low memory consumption. Node supports concurrent execution of business logic based on an asynchronous I/O event model. Asynchronous I/O is important for event-driven programming because it prevents the application from getting blocked while waiting in an I/O operation.

The messages exchange is ensured by the MQ Telemetry Transport (MQTT) protocol [25]. MQTT protocol is a lightweight application layer protocol designed for resource-constrained devices. It runs over TCP/IP, or over other network protocols that provide ordered, lossless and bidirectional connections. It uses the publish/subscribe messaging system combined with the concept of topics to provide one-to-many message distribution. Similar to HTTP, MQTT works on top of a persistent TCP connection. However, HTTP it is not suited for sending a large number of small messages as it incurs huge overheads due to its header sizes. Http also requires synchronous handshakes to set up a connection which makes scaling to a large number of subscribers difficult [26]. Unlike HTTP the headers of MQTT messages are small and the connection set up does not require a synchronous handshake which could support a range of 10 to 100 messages per second.

MQTT applies topic-based filtering of messages with a topic being part of each published message. The broker uses the topics to determine whether a subscribing client should receive the message or not. Clients can subscribe to as many topics as they are interested in.

7.2 | Evaluation

The evaluation of the presented solution was done from a resource use point of view to analyse whether the customizing option improvements would result in a better performance parameter. The parameters that were taken into account were memory use for stored data and the number of write operations in the local database.

To prove the benefits of the customized use of the gateway, we consider a scenario of controlling the temperature data of one patient with two different scenarios. The first is called fixed case, it is the standard case in which the data are collected with a unique interval of time. In the second case, the interval of data collection changes depending on the patient's score calculation.

In Figure 6, we illustrate a result of the self-adaptivity configuration. We support three levels of data storage frequency depending on the corresponding score. The orange signal presents the temperature measurements of a patient. The frequency of saving the sensed data changes when successful augmentation of temperature value is detected. The second signal in blue does not respect the self-adaptive algorithm contains unnecessary information for the first 7 hours and before the increase in the patient's temperature.

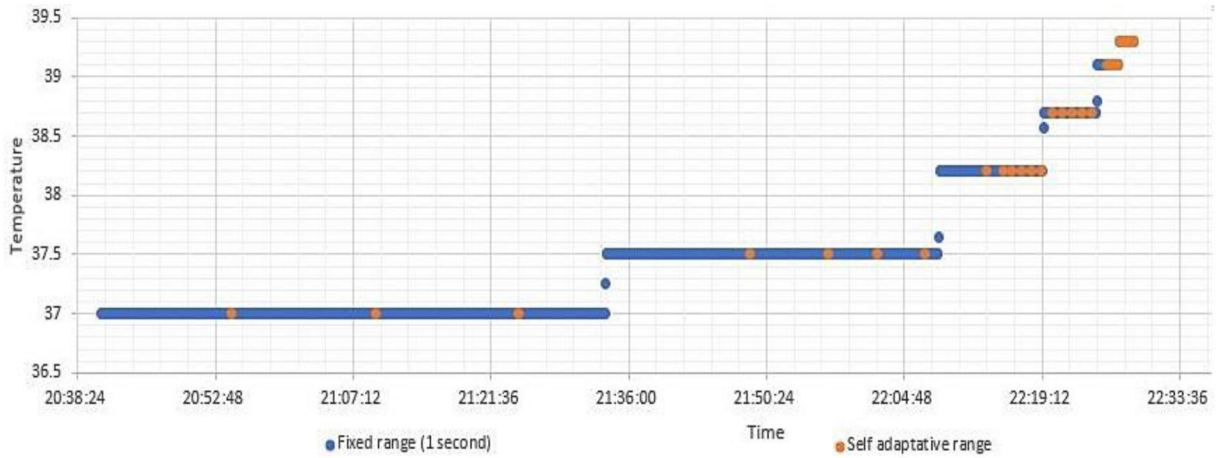


FIGURE 6 Self-adaptive configuration example

Our proposed system is predictive and responsive. Indeed, initially a medical staff stipulates a frequency of retrieves information concerning the state of a patient, however, the state of the latter can evolve (in one direction as in the other), which imposes the adaptation of our system by increasing or decreasing the measurement frequency. We calculated the temperature data size in the local database. We find that the saved data of the self-adaptive case present only 0.05% of the data saved with a fixed frequency of time.

Added to the self-adaptive algorithm advantage, we prove the benefits of the customized use of the gateway. We consider a scenario of controlling 10 patients in one day with three different scenarios. The first called fixed case, is the standard case in which all the patients' data are collected with a unique interval of time, the two other cases are custom cases where the interval of data collection is depending on the patient's situation and it is defined by the medical staff.

Table 1 illustrates the number of collected data by one device for one day in three different cases.

The evaluation results reflected in Figure 7 show that the use of a single data processing strategy in a fixed and non-custom way can result in an unnecessary use of gateway memory which obviously affects performance and the time reaction specially in emergency cases.

The example proves that the customizability of the smart healthcare system needs to be taken into account during the design stage and can be at one layer or more in the IoT-based solutions [27].

Added to that, we simulate five devices that send temperature values using the MQTT protocol. Two scenarios were made for this simulation. The first is to use a temperature interval equal to 1 s for the five devices. The second is to vary the sending intervals for each device (10 min, 5 min, 1 h, 30 min, 10 s).

The evaluation results show that the use of a single data processing strategy in a fixed and non-custom way can result an unnecessary use of gateway processor which obviously affects performance and the time reaction specially in emergency cases.

TABLE 1 Evaluation use case

Scenario	Interval of Time			Data
	1 s	1 min	1 h	
Fixed case	10	—	—	864,000
Custom case 1	3	5	2	266,448
Custom case 2	1	—	9	86,616

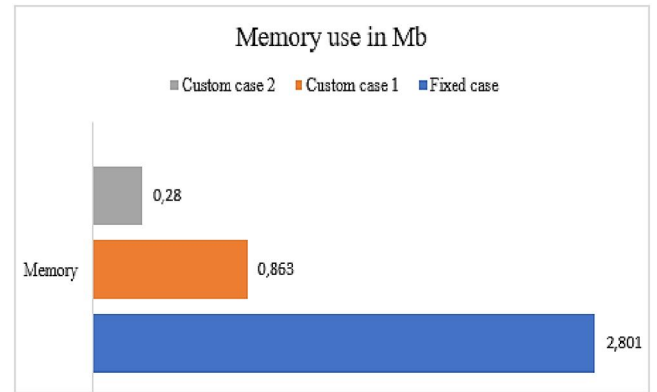


FIGURE 7 Memory use in megabytes

We are interested in measuring users disk writes. Comparing the results shown in Figure 8, we notice that the write operation to the InfluxDB database is reduced by 87.5% in the personalized configuration case.

Evaluation of the number of the created processes illustrated in Figure 9 shows a decrease of 82% in the number of processes using a personalized processing configuration. This significant gain is mainly due to the reduction in the number of data writing operations shown in Figure 8.

The Available memory presented in Figure 10, is estimated by the kernel, as the amount of RAM that can be used by user space processes without causing swaps. We notice a gain of two Mega for the case of personalized configuration.

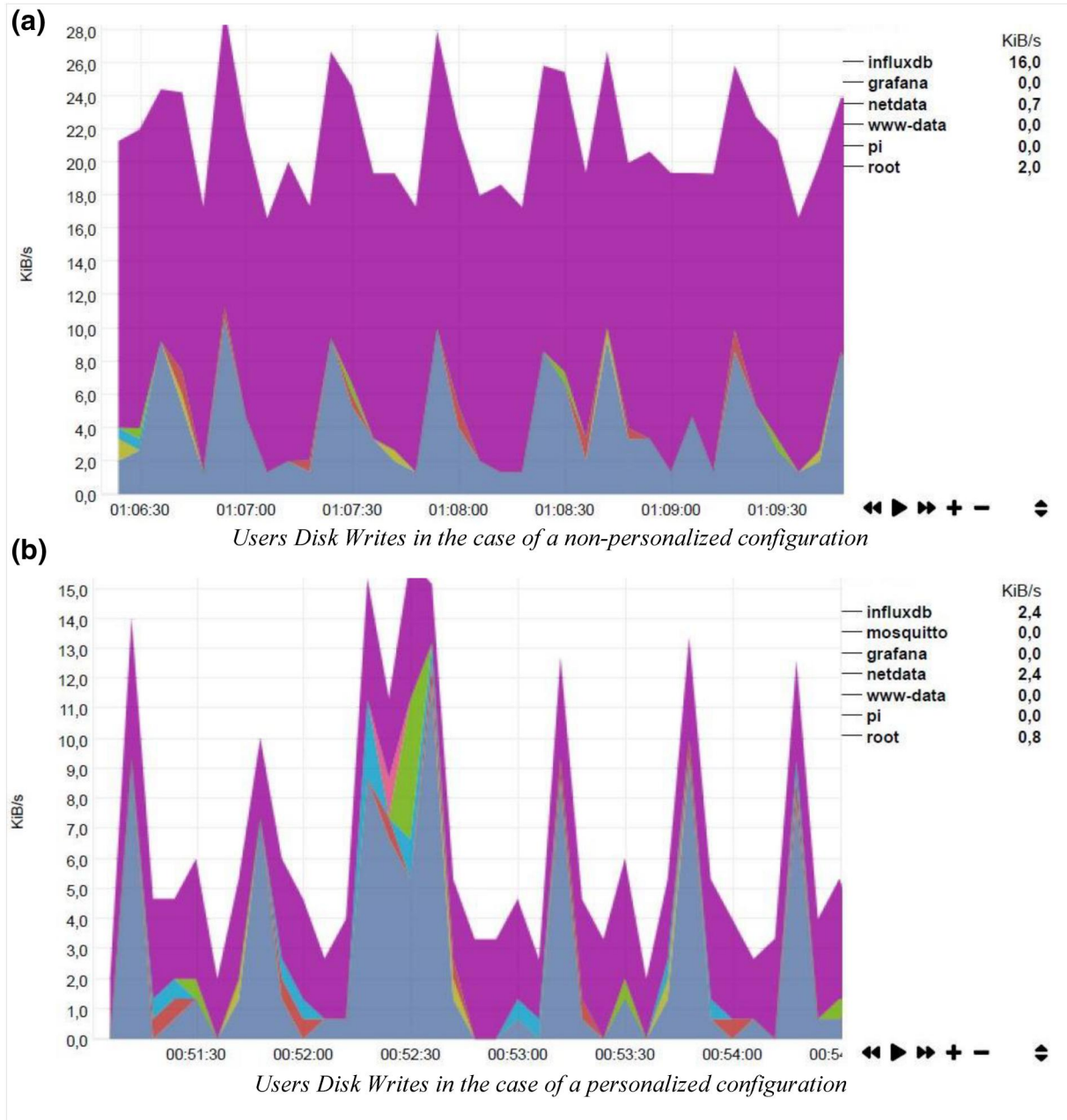


FIGURE 8 Users disk writes. (a) Users disk writes in the case of a non-personalized configuration. (b) Users disk writes in the case of a personalized configuration

8 | DISCUSSION

First, our solution takes the benefits of time and space decoupling of publish subscribe pattern.

- Space decoupling

The benefit of space decoupling is to enable many clients to subscribe to some data without affecting the producers of these data. The clients do not need to know the location of the medical devices. The only address a user application needs to know is the network address of the broker. The broker then routes the messages to the right applications based on the data

semantic. Decoupling the knowledge of the meaningful information from the knowledge of the physical topology is a simple and efficient way to monitor and control many devices located in many different networks. It ensures the system scalability and facilitates the integration of new devices.

- Time decoupling

Sending data by the medical devices and receiving it by the medical staff need not be timely coupled. The device can push messages to the gateway and terminate. The messages will be available for the medical staff any time later. The unreliability of the network edges and the heterogeneity of the connected

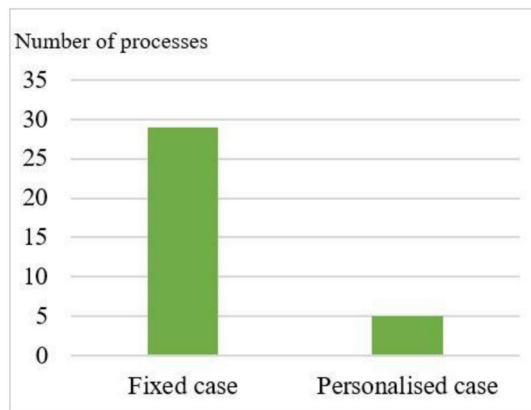


FIGURE 9 The number of the created processes

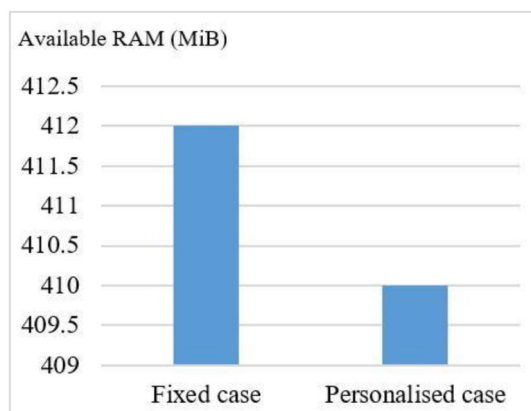


FIGURE 10 The Available RAM (MiB)

devices and applications make time decoupling an important property of the communication in an IoT application in medical context.

Second, the solution supports a self-adaptive approach to reduce the unnecessary processing of the sensed data. The results show that the responsiveness of our system can optimize energy consumption as well as reduce the need for material resources. This optimization increases the number of the controlled patients in each gateway and allocates more resources for the score calculation and notification process.

9 | CONCLUSION AND FUTURE WORK

We proposed an EWSs system that respects a risk evaluation approach named NEWS2. It provides a manual and self-adaptive configuration of the vital signs monitoring process depending on the patient's health status variation and the medical staff decisions. In case of risk detection, an intelligent notification mechanism is used to reduce the delay of the medical staff intervention. In future, we aim to use ontologies for more semantic interpretation of the collected data.

REFERENCES

1. Anzanpour, A., et al.: Internet of Health Things: toward intelligent vital signs monitoring in hospital wards. *J. Artif. Intell. Med.* 89, 61–69 (2018)
2. Downey, C.L., et al.: Strengths and limitations of early warning scores: a systematic review and narrative synthesis. *Int. J. Nurs. Stud.* 76, 106–119 (2017)
3. Le Lagadec, M.D., Dwyer, T.: Scoping review: the use of early warning systems for the identification of in-hospital patients at risk of deterioration. *J. Australian Critical Care.* 30(4), 211–218 (2017), issue
4. Anzanpour, A., et al., (eds.). In: Context-aware early warning system for in-home healthcare using internet-of-things, internet of things. *IoT Infrastructures, IoT360 2015. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, pp. 169. Springer (2016)
5. Yang, G., et al.: A health-IoT Platform based on the integration of intelligent packaging, unobtrusive bio-sensor, and intelligent medicine box. *IEEE Trans. Ind. Inform.* 10(4), 2180–2191 (2014)
6. Rahmani, A.M., et al.: “Exploiting smart e-Health gateways at the edge of healthcare Internet-of-Things: a fog computing approach”. *Future Generat. Comput. Syst.* 78(2), 641–658 (2018)
7. Farahani, B., et al.: Towards fog-driven IoT eHealth: promises and challenges of IoT in medicine and healthcare. *Future Generat. Comput. Syst.* 78(2), 659–676 (2018)
8. Zhang, Y., Qiu, M., Tsai, C.W., et al. Health-CPS: Healthcare cyber-physical system Assisted by cloud and big data. *IEEE Syst. J.*, 88–95 (2017)
9. Götzinger, M., et al.: Enhancing the self-aware early warning score system through fuzzified data reliability Assessment, Perego P. In: Rahmani, A., Taheri Nejad, N. (eds.) *Wireless Mobile Communication and Healthcare, MobiHealth 2017. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering*, vol. 247, pp. 3–11. Springer, Cham (2018)
10. Urban, R.W., et al.: Modified early warning system as a predictor for hospital admissions and previous visits in emergency departments. *Adv. Emerg. Nurs. J.* 37(4), 281–289 (2015)
11. Nguyen Gia, T., et al.: Low-cost fog-assisted health-care IoT system with energy-efficient sensor nodes, 13th International Wireless Communications and Mobile Computing Conference (IWCMC), Valencia, pp. 1765–1770 (2017)
12. Desai, P., Sheth, A., Anantharam, P.: Proc. of MS – International IEEE Conference on Mobile Services (2015). Semantic gateway as a service architecture for IoT interoperability
13. Smith, G., et al.: The ability of the National Early Warning Score (NEWS) to discriminate patients at risk of early cardiac arrest, unanticipated intensive care unit admission, and death. *Resuscitation.* 84(4), 465–470 (2013)
14. National early warning score (NEWS) 2, <https://www.rcplondon.ac.uk/projects/outputs/national-early-warning-score-news-2>, 2019
15. Du, B., et al.: KID model-driven things-edge-cloud computing paradigm for Traffic data as a service. *IEEE Netw.*, Jan.-Feb. 32(1), 34–41 (Jan-Feb 2018)
16. Mieronkoski, R., et al.: The Internet of Things for basic nursing care—a scoping review. *Int. J. Nurs. Stud.* 69, 78–90 (2017)
17. Aazam, M., Huh, E.N.: Fog computing and smart gateway based communication for cloud of things, pp. 464–470. International Conference on Future Internet of Things and Cloud, Barcelona (2014)
18. Razzaque, M.A., et al.: Middleware for internet of things: a survey. *IEEE Internet of Things J.*, Feb. 3(1), 70–95 (Feb 2016)
19. Botta, A., et al.: Integration of cloud computing and Internet of things: a survey. *Future Generat. Comput. Syst.* 56, 684–700 (2016)
20. Ben Ida, I., Jemai, A., Loukil, A.: A survey on security of IoT in the context of eHealth and clouds, pp. 25–30. 11th International Design & Test Symposium (IDT) (2016)
21. Richardson, M., Wallace, S.: *Getting Started with Raspberry PI*. O'Reilly Media, Inc (2012)
22. Rudolf, C., “SQL: noSQL or newSQL—comparison and applicability for Smart Spaces”. *Network Architectures and Services* (2017)

23. Light, R.A.: Mosquitto: server and client implementation of the MQTT protocol. *J. Open Source Softw.* 2, 13 (May 2017)
24. Tilkov, S., Vinoski, S.: Node.js: using JavaScript to build high-performance network programs. *IEEE Internet Comput.*, Nov.-Dec. 14(6), 80–83 (Nov–Dec 2010)
25. Banks, A., Gupta, R.: MQTT version 3.1. 1, pp. 29. OASIS standar (2014)
26. Sen, S., Balasubramanian, A.: A highly resilient and scalable broker architecture for IoT applications, pp. 336–341. 10th International Conference on Communication Systems & Networks (COMSNETS), Bengaluru (2018)
27. Ren, J., et al.: Serving at the edge: a scalable IoT architecture based on transparent computing. *IEEE Netw.* 31(5), 96–105 (2017)

How to cite this article: Ben Ida I, Balti M, Chaabane S, Jemai A. Adaptative vital signs monitoring system based on the early warning score approach in smart hospital context. *IET Smart Cities*. 2021;3:16–28. <https://doi.org/10.1049/smc2.12004>