Journal of Computer Science

Original Research Paper

Rapid Prototyping as a Low-Cost Push Factor to Promote the Adoption of IoT Technology in the Healthcare: A Case Study

[1]Paolino Di Felice, [2]Gaetanino Paolone and [3]Ernano Scalzone

[1]*Department of Industrial and Information Engineering and Economics, University of L'Aquila, Aquila, Italy*
[2]*B2B S.r.l., Teramo, Italy*
[3]*Gruppo Ingenious S.r.l., Mosciano Sant'Angelo, Teramo, Italy*

Article history
Received: 28-12-2023
Revised: 29-04-2024
Accepted: 27-05-2024

Corresponding Author:
Paolino Di Felice
Department of Industrial and Information Engineering and Economics, University of L'Aquila, Aquila, Italy
Email: paolino.difelice@univaq.it

Abstract: The increasing demographic aging of people in most countries all over the world, raises the issue of continuously monitoring their health status. At present days there is a big mismatch between the provision of assistance to adults and the actual demand. With advances in low-cost wearable devices, patients are becoming the first source of health-related data of themselves. That is the reason why scholars largely agree that resorting to solutions based on the Internet of Things (IoT) is the best way to provide assistance to senior citizens. Unfortunately, most organizations are behind in the adoption of the IoT. Healthcare is no exception. Previous studies have pointed out that without the influence of executive management, companies are likely to resist IoT adoption. So, the gap to be filled for the implementation of satisfactory long-term care services involves addressing a double challenge: Motivate health managers to invest in the IoT technology and, at the same time, prove the effectiveness of this typology of solution to physicians. This research gives four contributions: (a) it suggests the adoption of rapid prototyping as a tool to arouse interest in healthcare stakeholders; (b) it lists the features that IoT applications for the monitoring of remote patients must possess and which, therefore, must be first implemented in the rapid prototype; (c) it proposes ThingsBoard as the best candidate to build a rapid prototype; (d) it develops a case study that demonstrates that ThingsBoard simplifies and streamlines the development process of prototypes, hence making it a cost-effective solution for IoT rapid prototyping.

Keywords: Healthcare, Long-Term Care, Internet of Things, IoT Platform, Rapid Prototyping, ThingsBoard

Introduction

This section is structured in terms of five sub-sections (Fig. 1). They are presented in the following starting from a sub-section called "long-term care" and moving clockwise.

Long-Term Care

The 53 member states in the WHO1 European Region have agreed on the Health 2020 common policy framework. Their shared goals are to "improve the health and well-being of populations, reduce health inequalities, strengthen public health, and ensure people-centered health systems that are universal, equitable, sustainable and of high quality".

The good intentions just recalled conflict against the increasing demographic aging of the society of most countries of the world, which translates into a heavy financial and human burden for families and health centers (Social Protection Committee and European Commission, 2021).

Grages and Pfau-Effinger introduced the notion of a "care gap" as "a full or partial lack of provision of different forms of LTC for people with care needs" (Grages and Pfau-Effinger, 2022). Their study pointed out that in Europe there is a remarkable mismatch between the actual provision of LTC services and older persons' demand for such services. This situation takes place in most states all over the world. This is the case, for example, in China (Eggers and Xu, 2024). According to the findings in (Grages and Pfau-Effinger, 2022), "older people [...] are exposed to particularly high risks associated with care needs due to their increased vulnerability." Table 1 shows that Italians aged 85+ are 1,94 times more likely to be in need of care than persons between 64 and 74 of age.

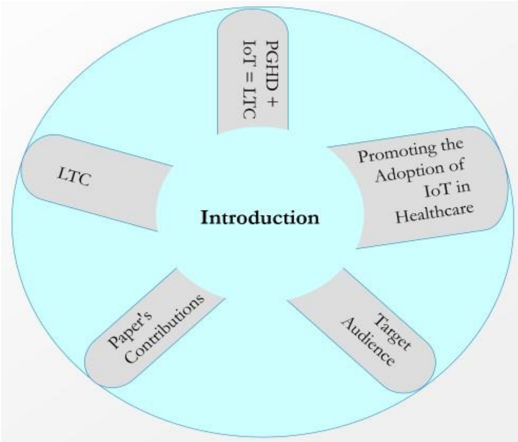

Fig. 1: Sub-sections in this section

Table 1: Differences in LTC needs for different age groups in Italy (source Grages and Pfau-Effinger, 2022)

65-74 Years	75-84 Years	Over 85 Years
40.8	63.1	79.3

LTC is usually understood as assistance to fragile subjects (that is, people with disabilities and elderlies) in their daily activities. Proof of this is the fact that LTC programs talk about family care (carried out either by their relatives or by employing low-cost migrant care workers this happens frequently in Italy (Grages and Pfau-Effinger, 2022)) and extra-familial care (usually nursing care) (Grages and Pfau-Effinger, 2022). Fig. 2 shows the places of assistance to fragile citizens (home and hospital) and the involved stakeholders (relatives, nurses, and physicians). The physician is located in between the patient's home and hospital to denote that its role is relevant in both scenarios. He is responsible for taking under constant monitoring the state of health of fragile persons.

PGHD + IoT = LTC

With advances in mobile health technologies (including activity trackers and other sensors), patients are able to generate continuous data Patient-generated Health Data (PGHD) are distinct from data generated in clinical settings (including health history, symptoms, biometric data, treatment history, lifestyle choices, etc.). PGHD are fundamental because they integrate what can emerge during visits with data collected 24 h a day and are not subject to the sensations and anxieties that patients often experience during clinic visits. PGHDs are the best candidates to integrate the health information gathered during patient visits and routinely stored inside the standard Electronic Patient Record (EPR), also called Electronic Health Record (EHR). Therefore, PGHD has the potential to enhance decision-making by providing valuable information that may not be captured during a routine care visit (Austin *et al*., 2020; Tiase *et al*., 2019; Demiris *et al*., 2019; Omoloja and Vundavalli, 2021; Kawu *et al*., 2023).

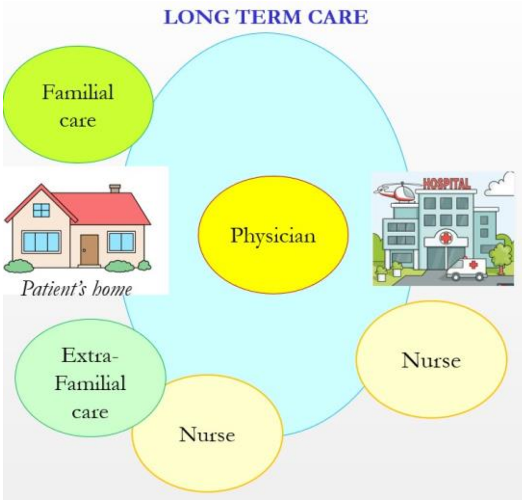

Fig. 2: Places of assistance to fragile citizens and the involved stakeholders

Previous studies have highlighted that the Internet of Things (IoT) technology allows the gathering, storage, and elaboration of PGHD, then allows the implementation of applications that make it possible to remotely monitor fragile subjects 24 h a day (Austin *et al*., 2020; Krishnamoorthy *et al*., 2023; Morello *et al*., 2022; Stavropoulos *et al*., 2020). Hence, the States can provide adequate LTC services to their citizens by resorting to solutions based on such a technology. It has been pointed out that IoT systems that implement remote healthcare services aim to improve trust, reliability, and cost-effectiveness of the overall service delivery (Abugabah and Nizamuddin, 2020). The containment of the cost of LTC programs is a precondition for healthcare sustainability (Mutingi and Mbohwa, 2014).

Promoting the Adoption of IoT in Healthcare

Several organizations are behind in the adoption of the IoT. Healthcare is no exception. In fact, the deployment of next-generation healthcare applications using the IoT (enhanced with Machine Learning, Edge/Fog Computing, and Blockchain technologies) is still far away (Krishnamoorthy *et al*., 2023; Liu *et al*., 2019). Olushola (2019) explored the following seven factors affecting IoT adoption: Executive management support, firm size, regulatory support, security concerns, technology readiness, compatibility, and complexity. "The executive management support holds a critical function in IoT adoption [...]. Without the influence of executive management, the company is likely to resist IoT adoption." (Olushola, 2019). In light of the findings just recalled, the point of view expressed in this study is that a way to break this deadlock consists of promoting the adoption of the IoT technology in the healthcare sector through campaigns consisting of demos of basic IoT applications, to be presented to health managers and physicians in order to spark their interest towards the innovation.

Target Audience

This research addresses IT firms of any size. Small and Medium-sized Enterprises (SMEs) are the global software industry's dominant force. Unfortunately, they suffer from time and cost overruns much more than big firms (Majchrowski *et al*., 2016). So, the best strategy for SMEs to be competitive is to use as much as possible open-source IT tools.

It was remarked that developing IoT applications is intrinsically difficult due to the many critical issues that this activity poses (Gavrilović and Mishra, 2021; Dias *et al*., 2022). From a software engineering perspective, IoT applications execute on a network of heterogeneous devices (e.g., sensors, actuators), operate in dynamic environments and they can interrupt their service without notice. Privacy, security, and performance are further relevant challenges for these applications. Therefore, it seems unlikely to hypothesize that SMEs and VSEs IT firms will develop IoT applications for the healthcare sector before receiving a formal assignment. On the other hand, it is equally unlikely to believe that the stakeholders of the healthcare sector will venture into investments in IoT technology without some persuasive demonstration of the benefits that could derive from such an investment.

Paper's Contributions

This study makes the following contributions:

- It suggests the adoption of rapid prototyping as a tool to arouse interest in healthcare stakeholders, building on a previous study (dos Santos *et al*., 2021)
- It lists the features that IoT applications for the monitoring of remote patients must possess and which, therefore, must be first implemented in the rapid prototype, in order to be adequately underlined during its presentation to the healthcare stakeholder
- It proposes ThingsBoard as the best candidate to build a rapid prototype, in light of the findings of a recent systematic mapping study (Di Felice, 2023a) that has elected ThingsBoard as the most mature open-source IoT platform
- It develops a case study that demonstrates that ThingsBoard simplifies and streamlines the development process of prototypes, hence making it a cost-effective solution for IoT rapid prototyping. The built prototype falls into the domain of the monitoring of remote patients. IT experts willing to enter into the IoT world can repeat by themselves our experiment. Then, they could start their own journey in the rapid prototyping of IoT applications to be illustrated to healthcare stakeholders

Materials and Methods

This section is structured in three sub-sections. They constitute the background of our study.

Software Prototyping

Prototyping in software development is a largely used paradigm. Both software engineers and stakeholders like it. A prototype simulates the product to be created and developed to test its (basic) functionality and/or the user interface/experience. Today, multiple taxonomies of prototypes are available. For instance, Wikipedia, CodiLime and Shenzhen LT Century Prototype Co., Ltd mention the following four categories: Rapid, Evolutionary, Incremental, and Extreme; while (dos Santos *et al*., 2021) mention the following: Low fidelity, high fidelity, vertical, and horizontal.

The underlying assumption of dos Santos *et al*. (2021) is that the adoption of prototypes in the workflow of product development can help the healthcare industry boost the engagement of stakeholders. To assess the soundness of such a claim, the authors carried out a systematic literature review devoted to investigating the adoption of different categories of prototypes in the development of health products. The articles taken into account in the study comprise 46 cases selected from Scopus and Web of Science databases. The outcome of the survey was that low-fidelity prototypes were used in 65% of cases (i.e., in 30 out of 46 cases). This category of prototype corresponds to a product skeleton with limitations on functionality and user interface, whose goal is to encourage users to provide feedback regarding the product concept. Low-fidelity prototypes support dissemination and information more than testing and training. Low-fidelity prototypes and rapid prototypes are equivalent. Within our paper, the latter naming is used.

The three columns of Table 2 collect the findings from (dos Santos *et al*., 2021). The italics sentences denote the objectives that can be achieved by adopting a rapid prototype before the development of an IoT application is started.

Desirable Features of the Rapid Prototype

The list of features that an IoT application for the monitoring of remote patients has to expose are listed in the following. Those features have to be embedded in the rapid prototype as well and adequately emphasized during its presentation to the healthcare stakeholders.

Deployment: In IoT, handling a large number of devices is a rule more than an exception. This is the case of a hospital that aims to offer a high-quality LTC service to all the inhabitants of the municipality by adopting remote monitoring of elderlies.

User management-authorization: Security and privacy represent relevant risks in the healthcare environment. It has been remarked that both these factors negatively influence IoT adoption (Charyyev *et al*., 2021). The user management authorization feature allows customizing access rights and permissions of users (i.e., managers, physicians, nurses, and so on) within a healthcare facility (e.g., hospitals, clinics, medical offices, and so on) according to the category of data at hand.

DOI: 10.3844/jcssp.2024.1185.1194

Data processing: PGHD have to be stored into a database in order to be processed according to physicians' needs.

Cloud hosting support: The Cloud computational paradigm is the one that best adheres to the scheme for implementing an effective LTC service.

Data visualization: Visualizing data comprises representing the gathered data according to the requirements in order to monitor past and current events. Furthermore, it must be provided the possibility to interact with the devices over a central interface.

Scalability: The IoT application has to be able to handle a large number of devices maintaining the same level of performance.

ThingsBoard Overview

ThingsBoard is an open-source IoT platform for data collection, processing, visualization, and device management (dos Santos *et al*., 2021). It enables device connectivity via industry-standard IoT protocols. ThingsBoard supports both on-premise and cloud deployments. With more than 5000 ThingsBoard servers running all over the world, ThingsBoard can run in production on AWS, Azure, and private data centers. Alternatively, it is possible to launch ThingsBoard in a private network without internet access. The ThingsBoard documentation (2023) lists the supported operating systems.

Architecture

The diagram of Fig. 3 shows the ThingsBoard's basic components and their interfaces. Following, a brief overview of these components is provided, focusing on the aspects relevant to the Case study.

Excellent documentation is available on the platform website. This is one of the relevant features of ThingsBoard in absolute terms, but also in comparison with most open-source IoT tools today available.

Transport Layer

ThingsBoard grants APIs for the common protocols (e.g., HTTP, MQTT, LwM2M, CoAP, and SNMP) suitable to work with IoT devices. Each API is provided by a dedicated server component that is part of the so-called "Transport Layer".

Message Queue

Messages from the devices are sent to the transport layer, then they are parsed and entered into the message queue. ThingsBoard supports multiple message queue implementations (e.g., kafka, AWS SQS, Azure Service Bus, and Google Pub/Sub).

ThingsBoard Core

The main ThingsBoard Core responsibility is handling the REST API calls.

Rule Engine

It is the heart of the system. Rule Engine is an easy-to-use framework for building highly customizable event-based workflows. No programming skill is required to use ThingsBoard. There are three main concepts behind ThingsBoard's rule engine:

- (Rule engine) message: Any incoming event (e.g., a data from a device)
- Rule node: A function that is executed on an incoming message. There are many different Node types that can filter, transform, or execute an action on the incoming message
- Rule chain: Nodes are connected with each other with relations. A Rule Chain is a logical group of rule nodes and their relations

Table 2: Findings from (dos Santos *et al*., 2021)

Prototype intents to gain user involvement	Prototype benefits for user involvement	Prototype as a communication medium
Consolidate a product idea	Refinement of the project	To facilitate communication
Negotiate the order of requirements	Project investigation	To evaluate user experience
Assess product functionality	Knowledge augmentation	

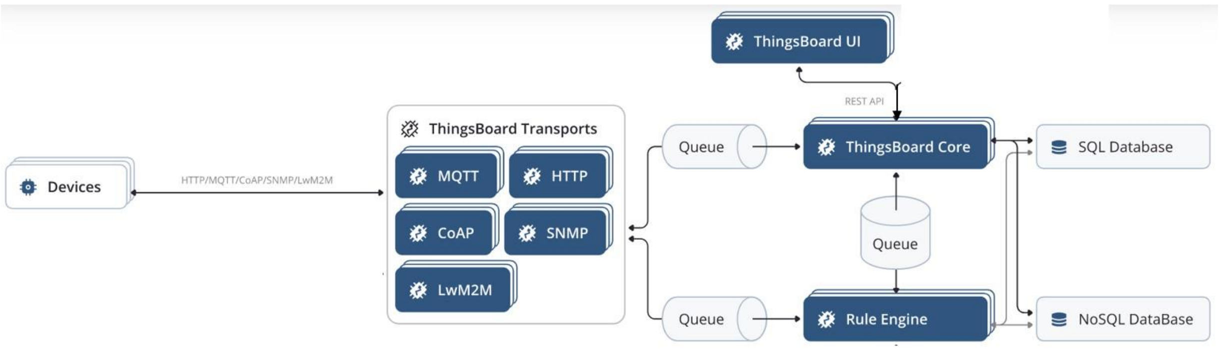

Fig. 3: Architecture of ThingsBoard (partial). The blue icons denote ThingsBoard's components

The root rule chain handles all incoming messages and may forward them to other rule chains for additional processing. A Tenant Administrator (Paolino Di Felice, in the case study) is able to define the root rule chain.

Storage

In ThingsBoard, entities (e.g., devices and customers) and telemetry data are stored in a PostgreSQL database (the pure SQL option). The database (called ThingsBoard) comprises 66 tables that collect metadata about the key concepts of a generic IoT application in addition to the involved telemetries; (b) alternatively, all entities are stored in a PostgreSQL database, while time series data are stored either as a Cassandra database or as a Timescale database (the two hybrid options).

User Interface

ThingsBoard features a lightweight web user interface.

Programming

ThingsBoard supports user-defined functions for data processing. The original programming language for the functions is JavaScript. It is popular, well-known, and simple. ThingsBoard Expression Language (TBEL) is the alternative to JavaScript. The reason for using TBEL (when possible) is because TBEL is lightweight and is much faster than Nashorn (the JavaScript engine).

The Features

In Held *et al.* (2022), the authors compared 7 open-source IoT platforms with respect to 14 distinct features. ThingsBoard supports the six desirable features listed above of the rapid prototype of an IoT healthcare application.

Architecture of a Remote Health Monitoring System

Figure 4 shows the components of the architecture of an IoT system for the remote monitoring of elderly citizens proposed in Sunehra and Siddireddygari (2020). In such a solution, ThingsBoard acts as a Web server responsible for the storage and visualization of the sensor values. Physicians can access the measures remotely by logging into ThingsBoard. The architecture of the IoT system proposed in Kadarina and Priambodo (2018) is conceptually identical to that in Sunehra and Siddireddygari (2020), nevertheless, both hardware and software make them different. In fact, Raspberry Pi and Arduino uno are used as microcontrollers, MQTT is used as a communication protocol between the sensors and ThingsBoard, while Python (plus a Paho Eclipse library) is used as a programming language. Barakat *et al.* (2021) is another study that proposed a remote IoT healthcare system to monitor the values of blood oxygen saturation and heart rate. The components of the implemented architecture are the following: Raspberry Pi 3 Model B (as a microcontroller), MAX30102 (as a pulse oximeter and heart-rate sensor), MQTT (as a communication protocol), and ThingsBoard as a visualization tool. In Narasimharao *et al.* (2023) confirmed the relevance of IoT remote health monitoring systems. Their solution adopts NodeMCU as a gateway to gather the user's health data, Raspberry Pi 4 as a central processing unit of the data received from the gateway, and ThingsBoard as a visualization platform.

By resorting to ThingsBoard, it is possible to build a rapid prototype of a healthcare IoT remote monitoring system ignoring the first three blocks of the architecture of Fig. 4. In fact, ThingsBoard allows to creation of virtual devices and generates telemetries. The simplification is very significant because it allows the IT developer to disregard both the hardware and the software necessary for the effective implementation of the IoT health monitoring system. The simplification, in turn, eliminates the costs connected to the purchase of the hardware, the procurement time and all the technical pitfalls to be overcome for their correct installation and usage. For example, if the Arduino 2560 microcontroller is adopted, it is a matter of resorting to both the Arduino Integrated Development Environment and the Embedded C as software tools to write the code that implements the IoT system.

Case Study

As said in the Introduction, the remote monitoring of fragile patients is fundamental to raising the quality of the LTC service of a nation. That is the reason why we believe that showing to the stakeholders of the healthcare domain this type of application can drive revenues to IT firms, in a relatively short time period after the marketing campaign.

The parameters we take under control in our example are blood oxygen saturation and heart rate (also called pulse rate). Controlling their values provides information about the heart condition. Many studies have taken into account those parameters from different perspectives (Tamam *et al*., 2018; Ali *et al*., 2020; Nemcova *et al*., 2020; Cao *et al*., 2020; Krizea *et al*., 2020; Lara-Cantón *et al*., 2022; Sharma *et al*., 2022; Totuk *et al*., 2023).

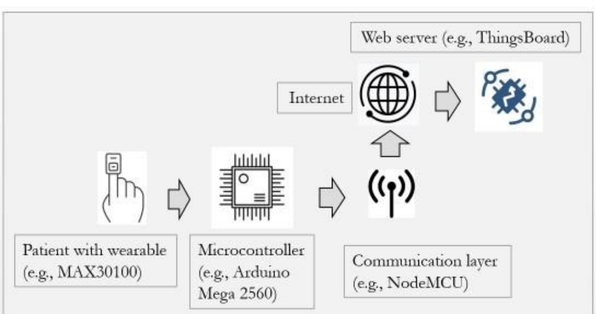

Fig. 4: The block diagram of a remote health monitoring system

DOI: 10.3844/jcssp.2024.1185.1194

The case study workflow comprises eight steps:

- Registration as a cloud ThingsBoard user
- Asset definition (optional)
- Device definition
- Pushing data from the (virtual) physical device to
- ThingsBoard
- Building a dashboard
- The setting of an alarm (optional)
- Generation of a continuous data stream
- Looking into the underlying database

The technicalities in the sequel are in brackets and are essential to repeat the experiment.

Registration: First of all it is necessary to register as a cloud ThingsBoard user in order to obtain the credentials to access the platform. The registered user (in this case Paolino Di Felice) becomes the Tenant Administrator of the application we are going to describe. A ThingsBoard Tenant is a separate business entity (that is an individual or an organization) that is allowed to define assets, devices, and customers.

Asset definition: For the use case at hand, we defined a Patient-Home asset (by clicking on the "+" icon that appears by selecting the Asset groups option of ThingsBoard). Default is the Asset profile that was given to such an asset.

Device definition: In ThingsBoard, devices (as they are called) are the basic IoT entities that can produce telemetry data. For the present use case, it was sufficient to create a single device able to sense the values of the blood oxygen saturation and the heart rate coming from a virtual physical device. (It is a matter of selecting the option All inside the item Device groups. Then, by clicking on the "+" symbol, the device will be created.)

ThingsBoard allows to establishment of a relationship between assets and devices. This descriptive feature is very useful in cases where there are a lot of devices and many assets. In our use case, the Patient-Home asset contains the Blood oxygen saturation and heart rate device.

Push Data from the Physical Device to ThingsBoard

A big simplification coming from the usage of ThingsBoard (in the development of prototypes of IoT applications) is being able to ignore the hardware and software characteristics of the electronic devices responsible for the collection (in real-time) of the actual measurements. Figure 5 shows (within the Windows command prompt) the HTTP POST command asking that data from the client be sent to ThingsBoard. The client simulates the missing physical sensor linked to the virtual device's blood oxygen saturation and heart rate, identified by the access token 8RuFood6sGcVdoUlhij3. In other words, the virtual device acts as a bridge towards ThingsBoard. In the command of Fig. 5, https://thingsboard. Cloud/ is the URL of the server running ThingsBoard cloud. The data sent (in the JSON format) are 94 and 89. Figure 6 shows that the message transmission succeeded.

Build a real-time end-user dashboard: ThingsBoard provides the ability to create and manage dashboards for the visualization of the generated telemetries. (As usual in ThingsBoard, it is sufficient to select the Dashboard groups menu option and click the "+" sign in the left upper right corner of the screen).

The next step in the process of construction of the kernel of the prototype of the IoT application is concerned with informing ThingsBoard about the device to be linked to the dashboard as a data source. This required the definition of what is called an Entity alias. We called it Biometric telemetries.

ThingsBoard offers a rich widget library to choose from. Widgets provide end-user functions (e.g., data visualization, remote device control, alarm management, etc.). Each widget has a data source. This is how the widget knows the data to be displayed. We selected the Chart widget. Figure 7 shows how the two measurements (of Fig. 5) are displayed on the dashboard.

```
C:\>curl -v POST --data "{"blood_oxygen_saturation":94,"heart_rate":89}"
https://thingsboard.cloud/api/v1/8RuFood6sGcVdoUlhij3/telemetry --header
"Content-Type:application/json"
```

Fig. 5: The POST request to publish telemetry data to the ThingsBoard server

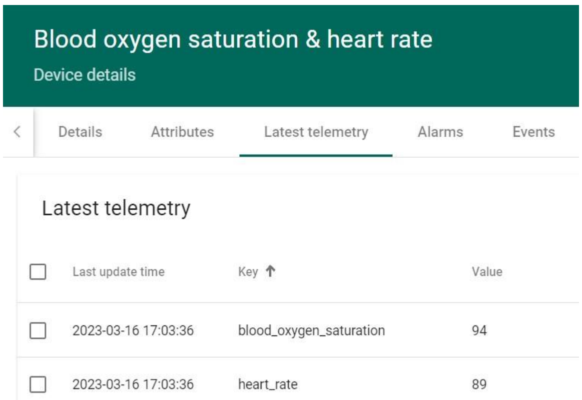

Fig. 6: The effect of the execution of the POST command

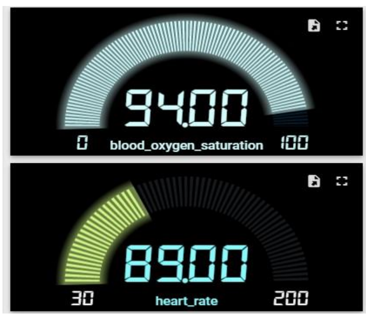

Fig. 7: Visualization of the telemetries

Set an alarm: In applications like that prototyped in this study, it is a best practice to implement alarms to be triggered in case the values of the telemetries exceed a critical threshold. Working with ThingsBoard, two actions are required to reach such a goal: (a) set an alert widget on the dashboard, and (b) create a rule that reads data in order to verify if the threshold value was exceeded or not. The first step consisted of entering the ThingsBoard dashboard (in our case: BOS&HR demo dashboard) in the editing mode and adding an alarm widget by selecting the Alarm widgets bundle. The second step concerned the creation of the rule responsible for triggering the alarm, when necessary. We used the alarm rules feature to raise the alarm when the \bloodoxygen_saturation" reading was less than 80 and, simultaneously, the \heartrate" reading was above 99 (a critical situation in an adult patient). (In order to do that, we went to the Rule chains menu option, selected the Script option left menu, and then wrote the logical condition shown in Fig. 8.)

Figure 9 shows the rule chain. As we see, it comprises also the create alarm and clear alarm nodes.

By posting the values \blood_oxygen _saturation":70 and \heart_rate":101 (same command as in Fig. 5), ThingsBoard reacted as shown in Fig. 10.

Generation of a continuous data stream: In the healthcare sector, to which the prototype refers, the telemetries concerning the remote patient must be read periodically and sent to the cloud server, where the physician can view them to assess the patient's conditions. In the use case, it was sufficient to set up a constant data stream from the Blood oxygen saturation and heart rate device to ThingsBoard. That required the creation of a data generator that emulated the physical device and sent messages (containing data) to the server. (Initially, the Rule chains option was selected in order to add a new Rule chain; then, a Generator rule node was added, responsible for the generation of messages with a configurable period. We set 60 sec. By clicking on the generator icon on the screen, it appeared a window containing the function Generate (prevMsg, prevMetadata, prevMsgType). We adjusted the body of such a predefined function as shown in Fig. 11. Math. random() generates random values from 85-100 and from 55-130, respectively, while Fixed (1) sets to 1 the number of decimal digits to be displayed. The Rule node applies the math function and saves the result into the message and into the hidden database, as well.

Figure 12 shows the snapshot of the visualization on the created dashboard of the continuous stream of telemetries generated by the Generate() function (Fig. 11). (By moving the mouse on the diagram, ThingsBoard shows the corresponding timestamp at which the measures refer to, Fig. 12. It is important to point out to the physicians that both these features will be part of the actual product).

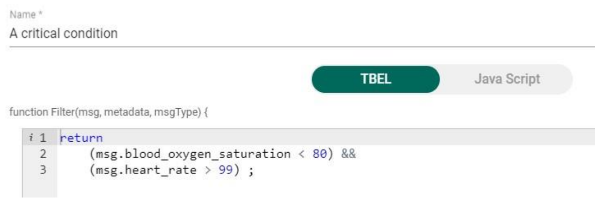

Fig. 8: Body of the Filter function

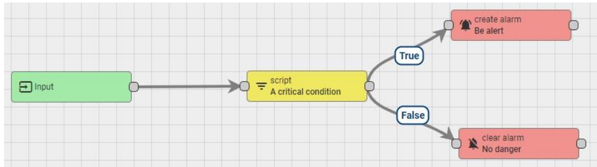

Fig. 9: The rule chain about the alarm

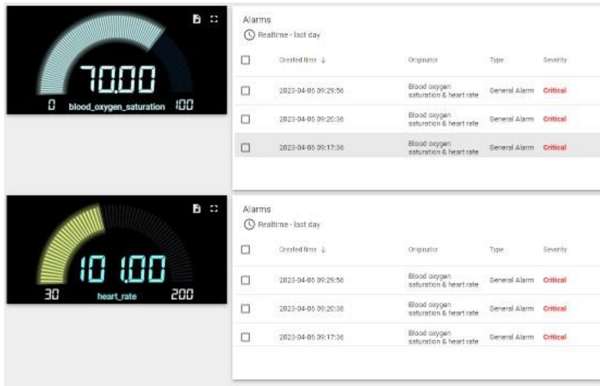

Fig. 10: Alarms of Critical severity issued by ThingsBoard

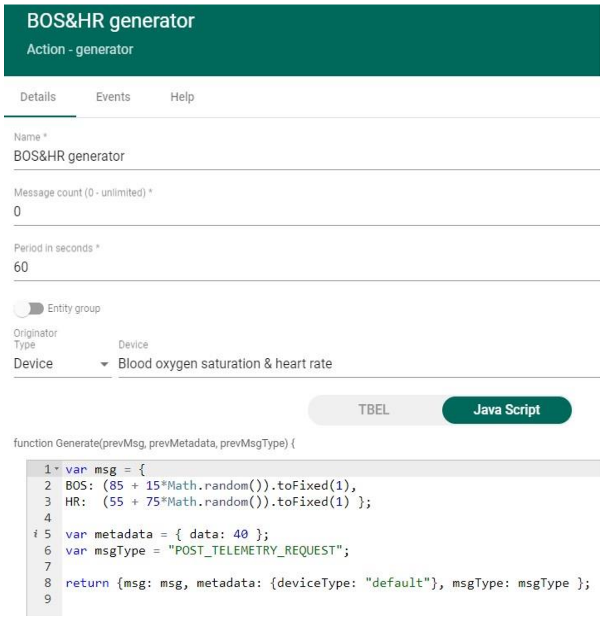

Fig. 11: The Java script code of function Generate(). TBEL does not support the Math. random() function

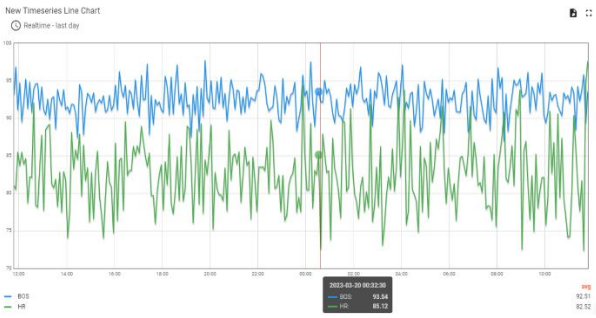

Fig. 12: Snapshot of the visualization of the continuous stream of telemetries

The ThingsBoard database: Di Felice (2023b) reports about the benefits coming from the alliance of the IoT and DBMS technologies when the final aim is to deliver an efficacious LTC service. When showing the physicians the prototype of the IoT application, the IT expert has to reserve time for illustrating the relevance of such a feature in the actual product. ThingsBoard facilitates this task since it manages the already mentioned ThingsBoard database. 8 out of the 66 tables that make up ThingsBoard are the core tables to focus on, namely: Tenant, asset, device, device profile, customer, tskvdictionary, and tskv. The latter table stores the telemetries. By querying tskv, it is possible to compute, for example, the summary statistics that, as pointed out in Di Felice (2023b), support physicians in performing “data-driven” evaluations of the physical conditions of remote patients.

Results and Discussion

The four contributions of the present research can be summarized as follows. Firstly, it suggests the adoption of rapid prototyping as a tool to arouse interest in healthcare stakeholders towards the adoption of IoT-based software applications able to raise the population well-being.

Secondly, the paper lists the features that IoT software applications for the monitoring of remote patients must possess and which, therefore, must be implemented in the rapid prototype.

Thirdly, the work proposes the ThingsBoard open-source IoT platform as the best candidate to build rapid prototypes because it supports the following relevant features: Deployment of IoT software applications can take place either on-premise or in the Cloud. The Cloud hosting support is the computational paradigm that best adheres to the scheme for implementing an effective LTC service; User management-authorization polices allow to protect security and privacy of patients; patient Data processing is possible according to physicians’ needs; customizable Data visualization features are available; Scalability of the IoT software application maintaining the same level of performance is guaranteed.

Lastly, a case study has been developed which demonstrates that ThingsBoard simplifies and streamlines the development process of prototypes, hence making it a cost-effective solution for IoT rapid prototyping.

Conclusion

Showing the healthcare stakeholders the skeleton of a prototype, as the one described above, the IT expert has the chance to emphasize the basic features that will be available in the future actual software application. Hopefully, the interaction between IT experts and potential future customers should generate interest and then desire, in them to invest in IoT software.

A brief discussion of directions for potential improvements follows. They will be part of our future work.

The case study focused on the explanation of the implementation process of the rapid prototype of the kernel of a remote monitoring IoT application by using the ThingsBoard platform. We plan to extend the application’s functionalities and then start collecting feedback from the potential stakeholders (both healthcare managers and physicians) in an actual healthcare environment. Such an effort will allow us to test the design concepts as well as assess the appeal of the prototype of the IoT application.

The case study addressed primarily the technical aspects when using ThingsBoard for building the prototype, but it didn’t delve into the potential challenges or limitations of the platform. Addressing any limitations or discussing potential drawbacks will provide a more comprehensive understanding of the solution. In addition, the accomplishment of such a step will offer the chance to match the final findings against the features of the ThingsBoard platform as evaluated by Held *et al*. (2022) and briefly recalled in a previous section.

Another line of investigation will be devoted to making a comparative analysis of ThingsBoard with some of the other six open-source IoT platforms mentioned in Held *et al*. (2022), in order to evaluate its effectiveness and uniqueness in the rapid prototyping of healthcare IoT applications. A broader perspective is necessary to achieve a balanced view of the solution’s advantages and disadvantages. Once again, the accomplishment of such a step will allow us to match the final findings against those reported by Held *et al*. (2022).

Acknowledgment

We appreciate the efforts of the Editorial team in reviewing and editing this study. Thank you to the Publisher for the support in the publication of this article.

Funding Information

This research was funded by B2B S.r.l., 64100 Teramo (Italy).

Author's Contributions

Paolino Di Felice: Conceived ideas, collected papers from different sources, designed the outline of the manuscript, and wrote the first draft of the paper.

Gaetanino Paolone and Ernano Scalzone: Reviewed the manuscript and provided critical feedback.

Ethics

This article is original and contains unpublished material.

References

Abugabah, A., & Nizamuddin, N. (2020). Smart Healthcare Ecosystem for Elderly Patient Care. *2020 43rd International Convention on Information, Communication and Electronic Technology (MIPRO)*, 365–370. https://doi.org/10.23919/mipro48935.2020.9245404

Ali, A. A., Hussein Ali, A., & Al-Askery, A. J. (2020). Design and Implementation of Smart E-Health System Based on Cloud Computing to Monitor the Vital Signs in Real-Time and Measurements Validation. *IOP Conference Series: Materials Science and Engineering*, *745*(1), 012097. https://doi.org/10.1088/1757-899x/745/1/012097

Austin, E., Lee, J. R., Amtmann, D., Bloch, R., Lawrence, S. O., McCall, D., Munson, S., & Lavallee, D. C. (2020). Use of patient-generated health data across healthcare settings: Implications for health systems. *JAMIA Open*, *3*(1), 70–76. https://doi.org/10.1093/jamiaopen/ooz065

Barakat, A. N., Ambark, T. M., & Bozed, K. A. (2021). Remote Healthcare Monitoring System using IoT platform. *The 7th International Conference on Engineering & MIS 2021*, 1–5. https://doi.org/10.1145/3492547.3492628

Cao, T., Tao, L., Liu, D., Wang, Q., & Sun, J. (2020). Design and Realization of Blood Oxygen and Heart Rate Sensor Nodes in Wireless Body Area Network. *2020 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, 469–473. https://doi.org/10.1109/icaica50127.2020.9182480

Charyyev, B., Mansouri, M., & Gunes, M. H. (2021). Modeling the Adoption of Internet of Things in Healthcare: A Systems Approach. *2021 IEEE International Symposium on Systems Engineering (ISSE)*, 1–8. https://doi.org/10.1109/isse51541.2021.9582493

Demiris, G., Iribarren, S. J., Sward, K., Lee, S., & Yang, R. (2019). Patient generated health data use in clinical practice: A systematic review. *Nursing Outlook*, *67*(4), 311–330. https://doi.org/10.1016/j.outlook.2019.04.005

Di Felice, P. (2023a). A Systematic Mapping Study about IoT Platforms. *The 4th International Electronic Conference on Applied Sciences*, 226. https://doi.org/10.3390/asec2023-15388

Di Felice, P. (2023b). IoT + DBMS = Periodic Summary of the Health Status of Remote Patients. *The 4th International Electronic Conference on Applied Sciences*, 257. https://doi.org/10.3390/asec2023-16371

Dias, J. P., Restivo, A., & Ferreira, H. S. (2022). Designing and constructing internet-of-Things systems: An overview of the ecosystem. *Internet of Things*, *19*, 100529. https://doi.org/10.1016/j.iot.2022.100529

dos Santos, T. B., Campese, C., Marcacini, R. M., Sinoara, R. A., Rezende, S. O., & Mascarenhas, J. (2021). Prototyping for user involvement activities: How to achieve major benefits. *CIRP Journal of Manufacturing Science and Technology*, *33*, 465-472. https://doi.org/10.1016/j.cirpj.2021.04.013

Eggers, T., & Xu, J. (2024). Integrating familial care and extra-familial care into a new long-term care policy for China: Examples from Germany's long-term care insurance. *International Journal of Social Welfare*, *33*(1), 42–60. https://doi.org/10.1111/ijsw.12581

Gavrilović, N., & Mishra, A. (2021). Software architecture of the internet of things (IoT) for smart city, healthcare and agriculture: analysis and improvement directions. *Journal of Ambient Intelligence and Humanized Computing*, *12*(1), 1315–1336. https://doi.org/10.1007/s12652-020-02197-3

Grages, G., & Pfau-Effinger, B. (2022). *Gaps in the provision of long-term care across Europe* (EUROSHIP Working Paper No. 14). Oslo Metropolitan University. https://doi.org/10.6084/m9.gshare.19328912

Held, F., Schauz, P., & Domaschka, J. (2022). A Systematic Comparison of IoT Middleware. In *Service-Oriented and Cloud Computing* (pp. 77–92). Springer. https://doi.org/10.1007/978-3-031-04718-3_5

Kadarina, T. M., & Priambodo, R. (2018). Monitoring heart rate and SpO_2 using Thingsboard IoT platform for mother and child preventive healthcare. *IOP Conference Series: Materials Science and Engineering*, *453*, 012028. https://doi.org/10.1088/1757-899x/453/1/012028

Kawu, A. A., Hederman, L., Doyle, J., & O'Sullivan, D. (2023). Patient generated health data and electronic health record integration, governance and socio-technical issues: A narrative review. *Informatics in Medicine Unlocked*, *37*, 101153. https://doi.org/10.1016/j.imu.2022.101153

Krishnamoorthy, S., Dua, A., & Gupta, S. (2023). Role of emerging technologies in future IoT-driven Healthcare 4.0 technologies: A survey, current challenges and future directions. *Journal of Ambient Intelligence and Humanized Computing*, *14*(1), 361–407. https://doi.org/10.1007/s12652-021-03302-w

Krizea, M., Gialelis, J., Kladas, A., Theodorou, G., Protopsaltis, G., & Koubias, S. (2020). Accurate Detection of Heart Rate and Blood Oxygen Saturation in Reflective Photoplethysmography. *2020 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, 1–4. https://doi.org/10.1109/isspit51521.2020.9408845

Lara-Cantón, I., Badurdeen, S., Dekker, J., Davis, P., Roberts, C., te Pas, A., & Vento, M. (2022). Oxygen saturation and heart rate in healthy term and late preterm infants with delayed cord clamping. *Pediatric Research*, 1–6. https://doi.org/10.1038/s41390-021-01805-y

Liu, Y., Zhang, L., Yang, Y., Zhou, L., Ren, L., Wang, F., Liu, R., Pang, Z., & Deen, M. J. (2019). A Novel Cloud-Based Framework for the Elderly Healthcare Services Using Digital Twin. *IEEE Access*, *7*, 49088–49101. https://doi.org/10.1109/access.2019.2909828

Majchrowski, A., Ponsard, C., Saadaoui, S., Flamand, J., & Deprez, J. (2016). Software development practices in small entities: an ISO29110-based survey. *Journal of Software: Evolution and Process*, *28*(11), 990–999. https://doi.org/10.1002/smr.1829

Morello, R., Ruffa, F., Jablonski, I., Fabbiano, L., & De Capua, C. (2022). An IoT based ECG system to diagnose cardiac pathologies for healthcare applications in smart cities. *Measurement*, *190*, 110685. https://doi.org/10.1016/j.measurement.2021.110685

Mutingi, M., & Mbohwa, C. (2014). Understanding sustainability in healthcare systems: A systems thinking perspective. *2014 IEEE International Conference on Industrial Engineering and Engineering Management*, 597–601. https://doi.org/10.1109/ieem.2014.7058708

Narasimharao, M., Swain, B., Nayak, P. P., & Bhuyan, S. (2023). Development of Real-Time Cloud Based Smart Remote Healthcare Monitoring System. In *Ambient Intelligence in Health Care* (pp. 217–224). Springer. https://doi.org/10.1007/978-981-19-6068-0_21

Nemcova, A., Jordanova, I., Varecka, M., Smisek, R., Marsanova, L., Smital, L., & Vitek, M. (2020). Monitoring of heart rate, blood oxygen saturation and blood pressure using a smartphone. *Biomedical Signal Processing and Control*, *59*, 101928. https://doi.org/10.1016/j.bspc.2020.101928

Olushola, O. B. (2019). Factors affecting IoT adoption. *IOSR Journal of Computer Engineering*, *21*(6), 19–24. https://doi.org/10.9790/0661-2106011924

Omoloja, A., & Vundavalli, S. (2021). Patient generated health data: Benefits and challenges. *Current Problems in Pediatric and Adolescent Health Care*, *51*(11), 101103. https://doi.org/10.1016/j.cppeds.2021.101103

Sharma, A., Tanwar, R. S., Singh, Y., Sharma, A., Daudra, S., Singal, G., Gadekallu, T. R., & Pancholi, S. (2022). Heart rate and blood pressure measurement based on photoplethysmogram signal using fast Fourier transform. *Computers and Electrical Engineering*, *101*, 108057. https://doi.org/10.1016/j.compeleceng.2022.108057

Social Protection Committee, & European Commission. (2021). *Long-Term Care Report Trends, challenges and opportunities in an ageing society*. European Union. https://doi.org/10.2767/677726

Stavropoulos, T. G., Papastergiou, A., Mpaltadoros, L., Nikolopoulos, S., & Kompatsiaris, I. (2020). IoT Wearable Sensors and Devices in Elderly Care: A Literature Review. *Sensors*, *20*(10), 2826. https://doi.org/10.3390/s20102826

Sunehra, D., & Siddireddygari, S. (2020). Patient health monitoring system using arduino mega 2560 and 630 Thingsboard server. *International Journal of Scientific and Technology Research*, *9*(3), 5020–5026.

Tamam, M. T., Taufiq, A. J., & Kusumawati, A. (2018). Design a system of measurement of heart rate, oxygen saturation in blood and body temperature with non-invasive method. *IOP Conference Series: Materials Science and Engineering*, *403*, 012038. https://doi.org/10.1088/1757-899x/403/1/012038

ThingsBoard documentation. (2023). https://github.com/thingsboard/

Tiase, V. L., Hull, W., McFarland, M. M., Sward, K. A., Del Fiol, G., Staes, C., Weir, C., & Cummins, M. R. (2019). Patient-generated health data and electronic health record integration: protocol for a scoping review. *BMJ Open*, *9*(12), e033073. https://doi.org/10.1136/bmjopen-2019-033073

Totuk, A., Bayramoglu, B., & Tayfur, I. (2023). Reliability of smartphone measurements of peripheral oxygen saturation and heart rate in hypotensive patients measurement of vital signs with smartphones. *Heliyon*, *9*(2), e13145. https://doi.org/10.1016/j.heliyon.2023.e13145