2 State of the Art

Being central to the work described in this dissertation, in this chapter a survey of pervasive healthcare applications is presented. In order to gain a greater understanding of which are the building blocks of a typical Internet of Things (IoT) system, a reference model for IoT systems is also discussed.

2.1 Internet of Things

2.1.1 Fundamentals of IoT

Internet of Things (or IoT) is an emerging communication paradigm, often hailed as the driver of the Fourth Industrial Revolution [1].

The definition of this concept has evolved over time with the development of other technologies such as data analytics, embedded systems, sensors, etc. Fundamentally, it can be described as the following [2]:

> IoT is a strategy supported on the development of networks of smart devices that exchange and process information through Machine-to-Machine (M2M) communications, usually based on the Internet Protocol (IP).

This technology enables ubiquitous systems to gather remarkable amounts of information regarding the surrounding environment, which can later be turned into insight through the usage of data fusion and data analytics tools, like Machine Learning.

In the specific context of healthcare, this technology can provide many benefits as it enables remote and continuous health monitoring [3, 4, 5]. It allows non-critical patients to be monitored from the comfort of their own houses, rather than in hospitals or clinics, reducing the strain on scarce hospital resources such as health professionals or beds. This is particularly beneficial to those who live in rural areas, with limited access to healthcare services. It enables elderly people and those with chronic diseases to have greater control over their own health, thus allowing them to live more independently. Moreover, with the automatization of medical procedures, these systems can make healthcare infrastructures more efficient and therefore lower the costs of healthcare [6, 7]. Particularly, in the realm of clinical research, by analyzing the data collected by these ubiquitous systems, it may be possible to find new relationships between certain pathologies and different physiological signals, such as variations in body temperature or heart rate [8]. These correlations, commonly referred to as biomarkers, can be used by these systems to assist clinical decisions, enabling novel predictive, prognostic, and diagnostic processes in healthcare.

2.2 A Reference Model for Pervasive Healthcare Applications

Reference models provide an abstract framework for designing systems and a set of commonly recommended practices for the application domain. It serves as a starting point in the design process, enabling the comprehension of complex systems by breaking them down into simple and distinct functional layers, while also defining some common terminology used in its domain.

In 2014, the IoT World Forum (IoTWF) architectural committee published an IoT architectural reference model, composed by seven layers as shown in Figure 2.1. This model [9] provides a simple and clean functional view into the different components of an IoT system, without narrowing the scope or locality of its components. However, from a hardware perspective, in this work we will focus on the most common approach taken by researchers, using 3 distinct components:

- **Endpoint** or **edge** nodes (corresponding to Layer 1), which interact with the physical world, capturing data.
- **Gateway** devices (Layer 3), which connect to one or multiple **edge** nodes, filtering and aggregating the data generated by these and communicating it to a central server;
- **Central** server (Layers 4-6), which is responsible for collecting, storing and analyzing the captured data in order to provide users with valuable insight;

While this model can be used to generically develop IoT systems for any industry (*e.g.* from agriculture to smart cities), in the context of the dissertation we will focus on pervasive healthcare applications.

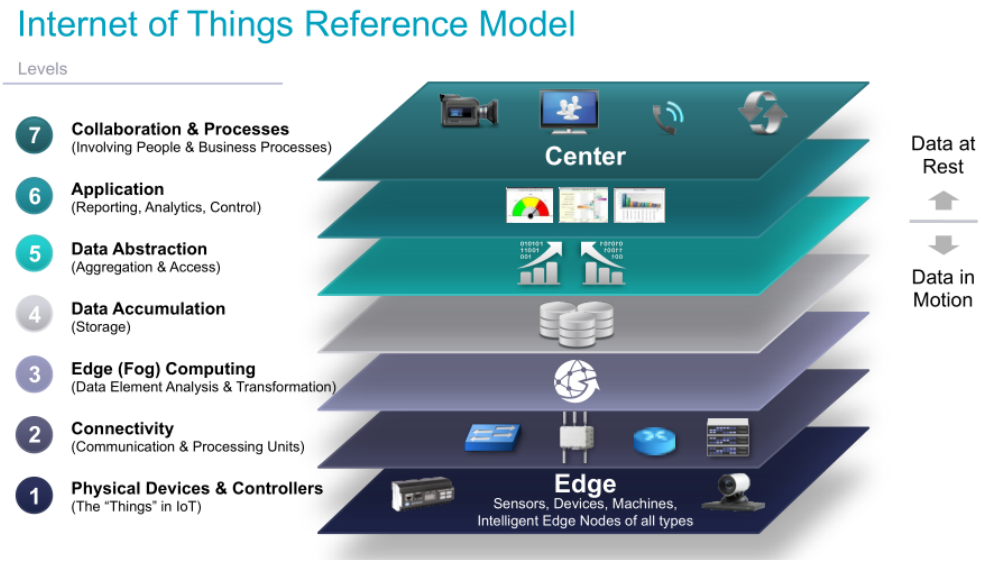

Internet of Things Reference Model
Levels
7 Collaboration & Processes
(Involving People & Business Processes)
6 Application
(Reporting, Analytics, Control)
5 Data Abstraction
(Aggregation & Access)
4 Data Accumulation
(Storage)
3 Edge (Fog) Computing
(Data Element Analysis & Transformation)
2 Connectivity
(Communication & Processing Units)
1 Physical Devices & Controllers
(The "Things" in IoT)
Center
Edge
Sensors, Devices, Machines,
Intelligent Edge Nodes of all types
Data at Rest
Data in Motion

Figure 2.1: IoT reference model published by IoTWF. Source: [9].

2.2.1 Layer 1: Physical Devices and Controllers

The first layer of the model [9] corresponds to the physical devices and controller layer. This layer houses the "things in the Internet of Things: the endpoint devices composed of sensors and actuators that perceive and interact with the physical world. Through those interactions, the devices generate data, which is then sent across the network for analysis and storage.

Wearable, wireless, and non-intrusive devices are viewed as one of the key components of IoT-based healthcare systems [2]. In recent years there has been remarkable progress on the development of wearable devices, driven by recent technological breakthroughs in the miniaturization of sensors and microfabrication processes [7, 6]. These devices allow patients to be monitored while retaining their mobility, increasing the comfort of these users. The drawback of this approach lies on the restrictions imposed on the devices. Due to the nature of this technology, most of these units require a portable energy source, which implies reduced memory, computation, and connectivity capabilities in order to minimize energy consumption and maximize their lifetime. Shorter lifetimes translate into higher maintenance costs, as these devices need to be replaced more often.

Another point to consider is the data requirements of the system, namely how much data is generated and which type of data is transmitted by each device. Some applications can include a single temperature sensor or heart rate sensor, while more complex systems can include pulse oximetry, electrocardiogram (ECG), respiration rate sensors, etc. [4]. From the literature [4, 10, 7, 11], the sensors used in these devices can be classified into three distinct categories based on the signals that can be extracted from them, as shown in Table 2.1:

- **Physiological Sensors**: used for evaluating the patients health condition.
- **Activity / Motion sensors**: used for detecting fall events, determining the patients location and the travelled distance, estimating the patients body posture, etc.
- **Environmental Sensors**: used for assessing environment conditions and possible hazards, *e.g.* gas leaks in a patients home or an industrial workplace [10].

Sensor Categories	**Examples**
Vital Signs Monitoring	Blood Pressure, ECG, PPG, Body Temperature, Respiratory Rate, Galvanic Skin Response, Pulse Oximetry, Glucose Level Sensors
Activity Monitoring	Accelerometer, Gyroscope, Magnetometer
Environmental Monitoring	Air Temperature, Barometer, Humidity, Gas Sensors

Table 2.1: Type of sensors commonly used in pervasive healthcare applications. Adapted from [11].

2.2.2 Layer 2: Connectivity

The second layer of the model focuses on connectivity, on linking the different components of our system, ensuring reliable and timely data transmissions. This includes all communications within the system, which can be divided into two categories: communications within the local network (*e.g.* between edge nodes and the gateway devices), and communications between the edge of the local network (*e.g.* gateway devices) and the central server.

Communication Protocols

Technology is designed with particular use cases in mind. They catalyze their development so each technology has its own advantages and disadvantages. For instance, short range wireless protocols are limited by transmission range, but long range protocols have a higher energy consumption, which may be unviable for networks with very constrained devices. Each protocol defines their own frame formats and communicates within certain frequency bands, some of which may require licenses. Using licensed frequency bands can provide a better performance as it ensures greater reliability since the network operator grants you exclusivity of frequency spectrum within a certain area.

From the literature, a set of key requirements that drive the decision of the communication protocols can be identified [2, 6, 7]:

- **Energy consumption**: For networks composed of energy constrained devices, the communication protocol should be lightweight and energy efficient in order to maximize the devices lifetime.

- **Latency**: Certain applications deal with time critical events, for example the detection of health emergencies [6]. In these cases, any delays in the communications can cause great detriment to the patients well-being, making it crucial to minimize them.

- **Reliability**: Depending on the critical nature of the data that is being communicated, the network stack may need to implement processes such as error-detection, retransmission or handshakes in the communications to ensure more robust transmissions, *e.g.* as implemented in TCP/IP based protocols. Generally, these features come at the cost of greater latency. Therefore, when choosing the communication protocol, a balance must be found between reliability and latency.

- **Security**: Security is one of the most important requirements of any system, but this is especially true for healthcare applications. Due to the sensitive nature of the information, it is crucial to secure it from malicious actors. Communication protocols must implement security mechanisms, such as encryption or data integrity verifications, that ensure the transmissions are not compromised in transit, thus denying third parties the ability to snoop or tamper the transmissions. This issue is studied in depth in [12].

- **Interoperability**: To ensure the interoperability of intrinsically different modules of the system it is imperative to choose protocols that are widely accepted and supported in the application domain. This also contributes to the longevity and maintenance of the system, as these will most likely remain supported for longer time periods.

- **Range**: The communication protocol must ensure the devices can communicate within the required transmission range.
- **Scalability**: These systems contain an enormous amount of devices, which must be uniquely identified. The communication protocol must ensure that every device in the network is addressable, and that performance is not severely impacted by the addition of new devices.
- **Throughput**: The communication protocol should ensure there is enough bandwidth to handle all communications within the designated transmission range. Even within similar technologies, this can vary wildly with the range [13].

Regarding communications within the local network, these generally have short transmission ranges [2]. Wearable devices can be often arranged in networks, aptly designated “Wireless Body Area Network”. The most widely adopted protocol is Bluetooth Low Energy (BLE), a low-energy version of the classic Bluetooth protocol [3, 10, 4]. ZigBee and Radio-frequency Identification (RFID) are also used in many systems in this domain, particularly in more asset tracking oriented applications [14, 6, 7]. Despite the absence of a universal specification for RFID, the most widely used standard is the EPCglobal Gen2 RFID [15], which is the standard considered in this work. For the sake of simplicity, we use EPC/RFID to designate it.

Out of the abovementioned technologies, BLE offers greater throughput, better security, and nearly the lowest energy consumption [16]. Table 2.2 shows a comparison between the different protocols.

	Bluetooth Low Energy	**EPC/RFID**	**ZigBee**
Band of operation	2.4 GHz	LF, HF, UHF, EHF	2.4 GHz
Communication	Bidirectional	Unidirectional (Bidirectional for Active tags)	Bidirectional
Topology	Point-to-Point, Piconet, Broadcast, Mesh	Point-to-Point	Mesh
Range	<100m	<10m, (100m for Active tags)	20m
Data rate	1Mb/s (up to 2Mb/s)	40kb/s	250kb/s
IP Stack	✗	✗	✓
Security Features	AES-128, Secure pairing prior to key exchange	✗	AES-128 (Optional), Network key shared across network, Optional link key to secure application layer communications

Table 2.2: Comparison between the more common communication protocols used within short range. Adapted from [2].

Regarding communications between the gateway devices and the central server, most researchers try to make use of existing infrastructure in order to facilitate the deployment of new systems. This means, that the communications between the gateway devices and the central server are often done through generic IP-based protocols such as Wi-Fi and Ethernet [7, 14, 4, 6].

Application Protocols

So far we have discussed the underlying networking technologies that link the devices in our system. But according to the OSI Model, many of these technologies do not define the application layer: how the devices communicate with each other, how the data is formatted, if there is a hierarchy within the network, etc. When considering networks composed of constrained devices, generic web-based protocols such as Hypertext Transfer Protocol (HTTP) may not be adequate for IoT applications, which prompts the development of novel lightweight messaging protocols suited for these systems. The most used application layer protocols in IoT systems are Message Queuing Telemetry Transport (MQTT) and Constrained Application Protocol (CoAP). Table 2.3 overviews these widely used protocols.

	MQTT	**CoAP**
Transport protocol	TCP/IP	UDP/IP
Messaging pattern	P-S (asynchronous)	R-R (synchronous)
Communication model	Many-to-many	One-to-one
Security	SSL/TLS (Optional)	DTLS
Strengths	TCP and Quality of Service (QoS), robust communications, easier to implement	Better for lossy networks, lower latency
Weaknesses	Higher overhead and energy consumption than CoAP	Not as reliable and less supported than MQTT

Table 2.3: Comparison between CoAP and MQTT protocols. Adapted from [13].

In [17], the authors compare these two protocols in greater length, along with the more commonly used web-based protocol Hypertext Transfer Protocol (HTTP). They analyze the latency in the communications (from the edge devices to remote servers) and the RAM usage in the devices for each protocol and for different data sources (respiration rate, oxygen saturation and heart rate signals) and found that CoAP presented the best overall performance. Nonetheless, all protocols had very low latencies (less than 1.5s) and low memory usage.

The authors indicate that MQTT might be more suitable when considering a certain messaging pattern — the “Publish/Subscribe (P-S) model. In this model, the devices that send messages, called “publishers”, communicate them to an intermediary message broker, through a “message topic (also called logical channels). The devices that wish to receive messages, called “subscribers”, can subscribe to these topics by requesting it to the message broker. Whenever a publisher sends a message, the broker broadcasts it to all devices that have subscribed to the selected topic.

CoAP uses a different messaging pattern, called “Request-Response”. In this paradigm, a device sends a request to receive certain data and the second responds to this request.

2.2.3 Layer 3: Edge (Fog) Computing

IoT systems may have hundreds or even thousands of sensors generating data multiple times per second, 24 hours per day, which may require an unsustainable amount of network and computing resources. Moreover, certain applications are time critical, where delays in communication can be very detrimental. To minimize these effects, it is crucial to initiate data processing as close to the edge of the network as possible. This paradigm is usually referred to as edge computing, when the data processing occurs at the endpoint devices, or fog computing, when it happens at the edge of local network, *e.g.* in gateway devices.

The third layer of the model defines how the system prepares the data for storage and higher level processing for the next layers. However, the endpoint or gateway devices often have limited computing capabilities, so the data processing is generally focused on preprocessing the data in real-time and handling more time critical events. More demanding and thorough data analysis is usually left to the central server.

The different processes applied at this stage can be summarized into four distinct categories:

- **Filtering**: Assessing if the data should be processed at a higher level.
- **Formatting**: Reformatting data to ensure consistent formats for higher-level processing.
- **Cleaning**: Reducing data to minimize the impact of data on the network and higher level processing systems.
- **Evaluation**: Determining whether data represents a threshold or alert. This is especially relevant for applications that deal with time critical events as seen in the previous section.

2.2.4 Layer 4: Data Accumulation

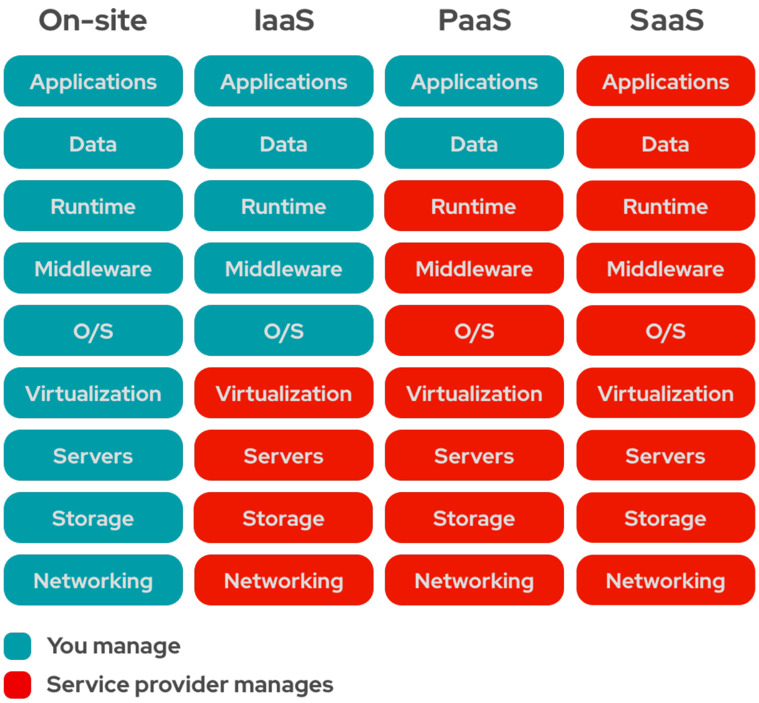
The data that is generated by the edge devices is propagated through the system, and eventually reaches the central server. Up to this point, model is event driven. However, most applications cannot make use of the data at the rate it is generated [13]. In the Data Accumulation layer, we need to define how the system captures the data and stores it, so it becomes usable for applications when needed, thus transiting from event to query-based processing.

As the devices continuously generate data, the system will require more and more resources in order to process and store all of this information, raising some concerns regarding how the data can be managed. In [3], the authors propose the usage of cloud platforms as a solution to these problems. This is made possible due to the elasticity in allocating, swiftly and inexpensively, computing and storage resources on-demand, adjusting itself to the needs of each application. Three distinct types of cloud services can be found:

- **Infrastructure as Service (IaaS)**: Provides control over the remote machine (composed of virtual or dedicated hardware), operative system and middleware. This approach gives system designers the highest level of flexibility over the infrastructure, but requires more maintenance.

- **Platform as a Service (PaaS)**: Provides a simple framework for developing applications, where the service provider manages the underlying infrastructure issues such as software updates and hardware maintenance.

- **Software as a Service (SaaS)**: Provides the finished applications to be used by the end users, in this case health workers, that enable them to work. A simple example is a web-based email service, such as Gmail or Microsoft Outlook.



On-site	IaaS	PaaS	SaaS
🟦 Applications	🟦 Applications	🟦 Applications	🟥 Applications
🟦 Data	🟦 Data	🟦 Data	🟥 Data
🟦 Runtime	🟦 Runtime	🟥 Runtime	🟥 Runtime
🟦 Middleware	🟦 Middleware	🟥 Middleware	🟥 Middleware
🟦 O/S	🟦 O/S	🟥 O/S	🟥 O/S
🟦 Virtualization	🟥 Virtualization	🟥 Virtualization	🟥 Virtualization
🟦 Servers	🟥 Servers	🟥 Servers	🟥 Servers
🟦 Storage	🟥 Storage	🟥 Storage	🟥 Storage
🟦 Networking	🟥 Networking	🟥 Networking	🟥 Networking

🟦 You manage

🟥 Service provider manages

Figure 2.2: Differences between the cloud offerings and on-premise solutions. Source: [18]

Nonetheless, security and privacy remain as key concerns for the implementation of cloud-based solutions. The information must remain accessible to authorized parties such as health-care providers, but the patients health data has to be kept private. To solve this, there are two commonly adopted features in the literature: access control policies and data encryption [2]. Access control policies define who can access the data, by authenticating them (validating the identity of the user) and by authorizing them (ensuring that the user has permissions to perform a given operation). Data encryption ensures that, even if the data is leaked, it is still unreadable to third parties, and therefore sensitive information remains secure and private.

Regarding storage solutions, most research works use traditional relational databases (SQL) as the means of storing data [14, 4, 6, 7]. However, as the data is very heterogenous and unstructured, the authors of [19] propose using non-relational databases (NoSQL) which can offer much greater flexibility as the schema in these databases is dynamic. Moreover, NoSQL type data stores outperform SQL databases as the data increases in volume [20]. Table 2.4 illustrates the differences between these two technologies.

	SQL	**NoSQL**
Type of database	Relational	Non-relational
Database model	Table-based database	Document-based databases, Key-value stores, graph stores, wide column stores
Data type	Appropriate for structured data	Appropriate for unstructured or semi-structured data
Schema	Strict schema	Dynamic schema
Query	Uses Standard Query Language (SQL), appropriate for complex query operations	Uses Unstructured Query Languages (*e.g.* UnQL), does not support complex query operations
Scalability	Vertical	Horizontal
Performance	Lower than NoSQL	Optimized for large datasets

Table 2.4: Comparison between SQL and NoSQL technologies.

2.2.5 Layer 5: Data Abstraction

In the previous layer, we have defined how the system captures the information. In some cases, the collection of data may require the development of multiple concurrent storage solutions, each using different technologies, resulting in a very complex environment. The purpose of this layer is to develop services that simplify how the applications access the data, to reconcile the different data stores and ensure the information is complete and consistent [9]. Applications can then interact with these databases through interfaces exposed by these services, designated Application Programming Interfaces (APIs).

An API is a computing interface that defines a set of rules that "explain how computers and applications communicate with one another, acting as an intermediary between these different components [21]. It defines which operations can be performed, how to request them, which are the accepted data types, etc. In this case, it decouples applications from the storage solutions, by encapsulating their functionality behind the interface. This ensures the modularity of the system as the applications become independent of whichever technologies are used in the data stores.

Understanding what and how information is shared within the healthcare domain is fundamental. As patients continuously move around the healthcare ecosystem, their health information must be available, discoverable and understandable to different entities (hospitals, laboratories, pharmacies, etc.). This prompts the digitization of medical files and the development of standards for exchanging these records instantly and securely to authorized parties [22], which are called Eletronic Health Records (EHRs). EHRs are the digital equivalent of a patient's paper-chart, they contain the patient's full medical history: previous diagnoses, treatment plans, test results, known allergies, among other details.

One of the most prominent standards for exchanging EHRs is Fast Healthcare Interoperability Resources (FHIR). FHIR is a standard developed by Health Level Seven International (HL7), which is a non-profit organization involved in the development of international healthcare informatics for over 20 years. FHIR builds upon previous data format standards like HL7 v2 and HL7 v3, and is becoming widely adopted within the healthcare industry [23]. This standard defines a lightweight RESTful framework using common data formats, like JSON and XML, so it can be readily integrated into lightweight web services, thus underlining its suitability for web-based platforms [24].

2.2.6 Layer 6: Application

The sixth layer corresponds to the application layer, where the system ingests the captured data, analyzes it and delivers the value to the end users. Users can then interact with the system through Graphical User Interfaces (UI), which provide different functionalities depending on the application. Some may show simple reports regarding the collected data [3, 4], and others may allow users to monitor and have greater control over the different components of the system.

As seen in an earlier section, Table 2.1 shows what kind of information is generally acquired in IoT healthcare applications. Using artificial intelligence, it is also possible to correlate all of this information to guide the clinical decisions from healthcare providers [24, 25, 26].

2.2.7 Layer 7: Collaboration and Processes

The information that is created by the IoT yields little value unless it prompts action, which requires people and processes (seventh layer). The objective is not the application — it is to empower people to work better and more efficiently. The sixth layer (Applications) provides business people the right insight, at the right time, so they can make the right decision. To do this people must be able to communicate and collaborate, which often requires multiple steps and transcends multiple applications [9]. However, this component of the system is beyond the scope of this work.

2.3 Survey on IoT Applications for Healthcare

We have thoroughly discussed how IoT systems are designed, but so far we have not discussed details of any specific implementation. This section presents an overview of IoT connected healthcare applications implemented in the literature, highlighting their strengths and weaknesses.

In [14], one of the first IoT applications for healthcare is described. The authors propose a real-time locating system (RTLS) using RFID tags called RFIDLocator. These tags are placed in hospital equipment, staff, patients and medical files and by using RFID readers placed in strategic locations around the hospital (*e.g.* entrance of rooms, handheld readers), it is possible to track the location of each object. When a RFID reader detects a RFID tag it communicates this information, using Wi-Fi, to a central server which stores it in a MySQL database. Healthcare workers can then view this information through a web application, which contains a location history of the tagged object. The authors show how RTLS systems can mitigate the risks of patient misidentification, loss or theft of assets and even drug counterfeiting. However, in this article, security and privacy issues are not discussed. Although not stated explicitly, communications between the RFID tags and the RFID readers are assumed to be unencrypted, which means "unethical individuals could snoop on people and surreptitiously collect data (...) without their knowledge", even after leaving the hospital if the tags are not removed. This raises serious privacy concerns, as the tags could contain private information that can be detrimental to the patients if revealed.

In [7], the authors propose a RTLS system that also monitors the patient's vital signs, using a small wristband which holds a low power device equipped with temperature, photoplethysmography (PPG), used to obtain the heart rate, and accelerometer sensors, used for detecting fall events. The system can also detect with 70% accuracy if the patient has fallen, sending an immediate message to the gateway, which will later alert the clinical staff to the emergency. The authors ran a pilot test within hospital premises which was well-received by the clinical staff who praised the system for its intuitiveness and non-intrusiveness, stating that it could be easily integrated into their current HIS. However, the authors pointed out some issues related to the usage of RFID tags with sensors for patient monitoring. The RFID reader powers to the RFID tags, and when using tags with sensors, the readers need to provide considerably more energy to the tags. The readers must be adjusted to provide enough power, but local regulations limit the transmission power. Regarding e-health standards, the authors did not discuss any protocols for exchanging data such as FHIR, which can undermine the integration of the system with existing HISs.

Wu et al. [4] develop a system which uses wearable sensor patches to monitor the patients' status. The wearable sensors transmit the different physiological signals (ECG, PPG and body temperature) to gateways using BLE, which can either by fixed (using a Raspberry Pi module) or mobile (using a smartphone app). The gateway exchanges data with the cloud through bridged MQTT brokers, after which it is stored in a MySQL database. The data is stored both in the cloud server and in the fixed gateway. The local users can interact with the system through a web-based user interface (UI) using the smartphone or other web browsers in the local area network. However, the usage of local data storage can cause data integrity issues as the system must ensure databases in both the server and gateways are synchronized at all times. This can undermine the scalability of the system, as the redundant data synchronization can become a performance bottleneck in the long term.

In [3], the authors proposed a IoT infrastructure that acquires real-time patient data from wearable sensors, using a cloud platform to handle all data processing and storage requirements. The authors developed a wearable device which takes the form of a sock, designated "CloudSensorSock". The CloudSensorSock acquires mobile data, through accelerometer and gyroscope sensors, vital data, through temperature and heartbeat sensors, and contextual information about the patient's environment using air quality (CO_2) sensors. It communicates with a mobile app through BLE, which acts as a gateway to the cloud server. The authors propose moving the data processing entirely to the cloud server as cloud platforms can scale to the needs of the application with little management and cost. However, this approach may not be viable for time critical applications. As discussed earlier, the latency in the communications between devices and remote servers may be too detrimental to the application, especially since the authors propose using this system for detecting fall events.

Recently, and motivated by the pandemic crisis, researchers from Institute of Systems and Robotics (ISR) in Lisbon developed a system called “e-CoVig” a low-cost solution for monitoring COVID-19 patients during the quarantine, as shown in Figure 2.3 [26]. The data acquisition is performed using a mobile app. To collect physiological data, the authors developed a specialized wearable device that communicates with the mobile app through BLE, recording pulse oximetry (SpO_2), heart rate, and temperature data. Alternatively, patients can use their own measuring devices, *e.g.* a thermometer, and manually insert the measurements or use Optical Character Recognition (OCR) to automate the in-app insertion of the values. The app can also be used to record audio snippets in order to detect cough and monitor respiratory activity. Unfortunately, the lack of e-health standards hinders its integration with external healthcare systems.

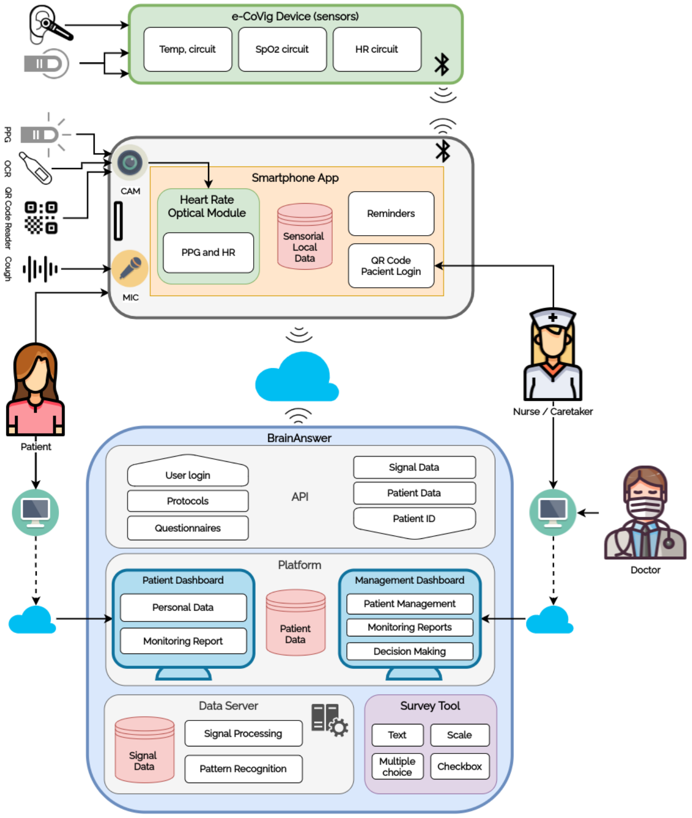


Figure 2.3: Overview of e-Covig’s system architecture. Source: [26].

2.3.1 Comparative Analysis

References	Measured Signals	Networking Protocols	Data Storage	e-Health Standards	Application Features	Security Features
[14]	✗	EPC/RFID, Wi-Fi	MySQL	None	RTLS	Unspecified Storage Encryption
[7]	Temperature, Heart Rate, Accelerometer	EPC/RFID, Wi-Fi	MySQL	None	RTLS, Fall Detection, Vital signs monitoring	AES-128, WPA-Personal
[4]	Temperature, Heart Rate, Accelerometer	BLE, Wi-Fi, MQTT	MySQL	None	RTLS, Fall Detection, Vital signs monitoring	AES-128
[3]	Temperature, Heart Rate, Accelerometer, CO_2 Sensor	BLE, Wi-Fi, GPRS/3G HTTP	MySQL	None	Fall Detection, Vital signs monitoring	AES
[26]	Temperature, Heart Rate, Pulse Oximetry, Respiration Rate	BLE Wi-Fi	Unknown	None	Fall Detection, Vital signs monitoring, Clinical decision support	Unknown

Table 2.5: Comparison between different pervasive healthcare applications.

2.3.2 Weaknesses of literature

Security and Privacy

From the literature, we find that many solutions secure communications between the devices using standard encryption algorithms like Advanced Encryption Standard (AES) [7, 4, 3]. However, very few discuss authentication and authorization processes [3, 12]. To ensure that no data is leaked to malicious authors, the networking protocols used must ensure these security properties.

Many web services currently use Transport Layer Security (TLS)[1]. This protocol ensures integrity, confidentiality and authentication as it combines public key cryptography to validate the identity of the communicating parties, symmetric-key algorithms to encrypt the transmissions and message integrity checks to ensure the transmissions are not tampered during transport. These properties make this protocol invaluable for secure communications over the web, and thus should be an integral component of IoT networks. Moreover, access control needs be considered. The systems are composed with many devices, which may have different levels of access level for each device. For example, limiting access to certain topics in MQTT, so that devices can only subscribe and publish messages in specific topics.

Interoperability

Despite recent efforts, interoperability is still an issue of IoT systems. Due to the lack of clear and concise industry standards and regulations, many manufacturers develop their own proprietary data formats and communication protocols, which hampers the integration of new resources since the systems are designed within closed ecosystems [17]. Moreover, the adoption of new systems can be often met with much objection from the clinical staff due to their mistrust of technology [27]. To facilitate the deployment of new healthcare systems in hospitals, these need to be integrated easily in existing HIS.

Fortunately, there are several international initiatives to promote the use of IoT in health in a standardized way, such as HIMSS (Healthcare Information and Management Systems Society) and the Personal Connected Health Alliance (PCHAlliance). PCHAlliance, for example, advocates the adoption of the Continua Design Guidelines (CDG), which facilitates the integration of personal health devices into health systems. These guidelines have been

[1] The Transport Layer Security (TLS) Protocol Version 1.3: https://tools.ietf.org/html/rfc8446

recognized by ITU (International Telecommunication Union) and the European Commission and are adopted by countries such as Denmark, Norway and the USA, among others [28]. These guidelines promote a series of e-health standards like FHIR which facilitate the exchange of information between systems, in order to ensure that the implementations become truly interoperable.

2.4 Statement of Contributions

After studying the different approaches taken by researchers, and in the context of the dissertation, we propose a novel fully modular IoT infrastructure that uses the FHIR standard in order to fully integrate the data into the existing HIS, Glintt GlobalCare. Other researchers in the Institute of Systems and Robotics (ISR) have already developed wearable devices, designated “biostickers”, using two different networking stacks — BLE and EPC/RFID. The system must ensure reliable and secure communications with the devices for each protocol. From a hardware perspective, the system is composed of 3 different components, as seen in Figure 2.4: the biostickers, that acquire the patient’s physiological signals, the “SmartBox”, which acts as a central node of the WBAN and aggregates the data, communicating it to the gateway, and the “Smart Gateway”, which serves as a fog server in order to mitigate latency and other computing issues and acts as the gateway to the HIS.

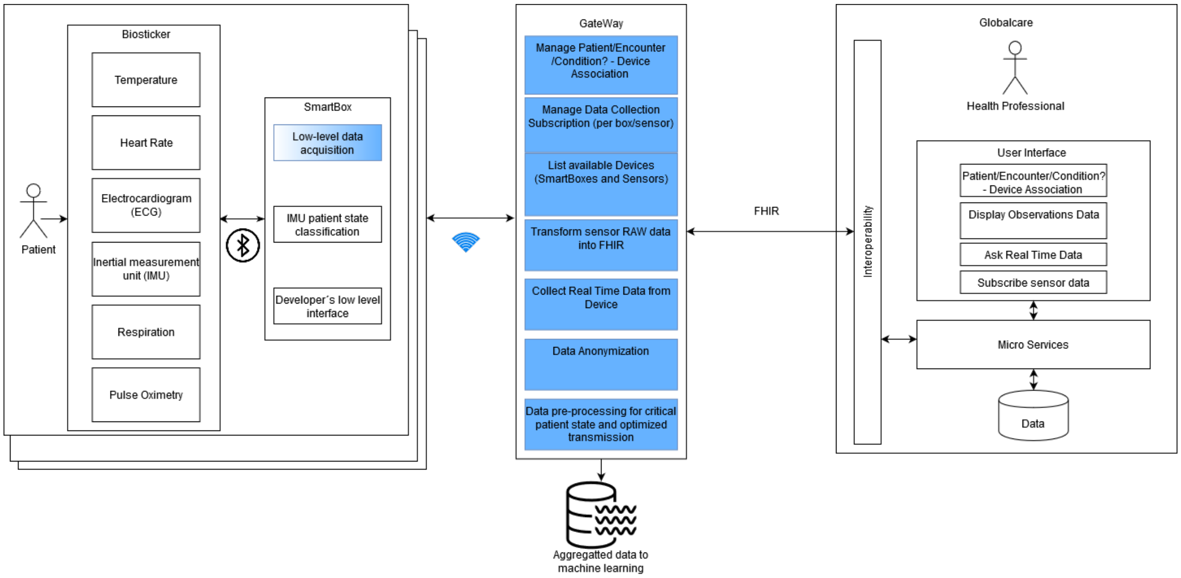

Biosticker
Temperature
Heart Rate
Electrocardiogram (ECG)
Inertial measurement unit (IMU)
Respiration
Pulse Oximetry
Patient
SmartBox
Low-level data acquisition
IMU patient state classification
Developer´s low level interface
GateWay
Manage Patient/Encounter /Condition? - Device Association
Manage Data Collection Subscription (per box/sensor)
List available Devices (SmartBoxes and Sensors)
Transform sensor RAW data into FHIR
Collect Real Time Data from Device
Data Anonymization
Data pre-processing for critical patient state and optimized transmission
Aggregatted data to machine learning
FHIR
Globalcare
Health Professional
Interoperability
User Interface
Patient/Encounter/Condition? - Device Association
Display Observations Data
Ask Real Time Data
Subscribe sensor data
Micro Services
Data

Figure 2.4: System architecture of the WoW project. The components highlighted in blue are discussed and implemented throughout this work.

For the work developed throughout the dissertation, we have set the following goals:

- Develop and deploy SmartBoxes embedded in hospital beds for data acquisition from biostickers attached to patients' skin:
  - Hardware evaluation of 2 different IoT kits (Raspberry Pi and Udoo Bolt).
  - Viability study of a disruptive battery-less RFID data acquisition compared to a BLE acquisition.
  - Implementation of reliable and secure RFID data acquisition and general communication between the biosticker and the Smart Boxes.
  - Selection of hardware components and assembly of SmartBox prototype;
- Establishment of data integration pipelines using MQTT and management of the multiple SmartBoxes in the Smart Gateway;
- Development of a FHIR API layer to integrate the proposed system in the GlobalCare HIS.
- Evaluation of the performance of the proposed system through controlled lab tests and later deployment hospital trials within the WoW project.
- Evaluate the need for data analysis to extract relevant user profiles/anomalies for long-term biomonitoring data of multiple anonymous users.

2.5 Summary

In this chapter, we have discussed the importance of IoT systems, how these are composed and how these can bring value to healthcare providers. After analyzing the different systems in the literature, we have defined a set of criteria to guide the development of our own implementation.

In the next chapter, we begin with the hardware evaluation of the different IoT kits and the implementation of secure communications between the biostickers and the SmartBoxes.

Bibliography

[1] G. Aceto, V. Persico, and A. Pescapé, "Industry 4.0 and Health: Internet of Things, Big Data, and Cloud Computing for Healthcare 4.0," *Journal of Industrial Information Integration*, vol. 18, no. February, p. 100129, 2020.

[2] S. B. Baker, W. Xiang, and I. Atkinson, "Internet of Things for Smart Healthcare: Technologies, Challenges, and Opportunities," *IEEE Access*, vol. 5, pp. 26521–26544, 2017.

[3] C. Doukas and I. Maglogiannis, "Bringing IoT and Cloud Computing towards Pervasive Healthcare," in *2012 Sixth International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*, pp. 922–926, IEEE, jul 2012.

[4] T. Wu, F. Wu, C. Qiu, J. M. Redoute, and M. R. Yuce, "A Rigid-Flex Wearable Health Monitoring Sensor Patch for IoT-Connected Healthcare Applications," *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 6932–6945, 2020.

[5] Yuan Jie Fan, Yue Hong Yin, Li Da Xu, Yan Zeng, and Fan Wu, "IoT-Based Smart Rehabilitation System," *IEEE Transactions on Industrial Informatics*, vol. 10, pp. 1568–1577, may 2014.

[6] L. Catarinucci, D. de Donno, L. Mainetti, L. Palano, L. Patrono, M. L. Stefanizzi, and L. Tarricone, "An IoT-Aware Architecture for Smart Healthcare Systems," *IEEE Internet of Things Journal*, vol. 2, pp. 515–526, dec 2015.

[7] T. Adame, A. Bel, A. Carreras, J. Melià-Seguí, M. Oliver, and R. Pous, "CUIDATS: An RFID–WSN hybrid monitoring system for smart health care environments," *Future Generation Computer Systems*, vol. 78, pp. 602–615, jan 2018.

[8] E. Choi, M. T. Bahadori, A. Schuetz, W. F. Stewart, and J. Sun, "Doctor AI: Predicting Clinical Events via Recurrent Neural Networks.," *JMLR workshop and conference proceedings*, vol. 56, pp. 301–318, 2016.

[9] Cisco, "The Internet of Things Reference Model," 2014.

[10] F. Wu, T. Wu, and M. R. Yuce, "Design and Implementation of a Wearable Sensor Network System for IoT-Connected Safety and Health Applications," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, pp. 87–90, IEEE, apr 2019.

[11] L. Minh Dang, M. J. Piran, D. Han, K. Min, and H. Moon, "A survey on internet of things and cloud computing for healthcare," *Electronics (Switzerland)*, vol. 8, no. 7, pp. 1–49, 2019.

[12] P. Gope and T. Hwang, "BSN-Care: A Secure IoT-Based Modern Healthcare System Using Body Sensor Network," *IEEE Sensors Journal*, vol. 16, no. 5, pp. 1368–1376, 2016.

[13] D. Hanes, G. Salgueiro, P. Grossetete, R. Barton, and J. Henry, *IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things.* Cisco Press, 1st ed., 2017.

[14] P. Fuhrer and D. Guinard, "Building a smart hospital using RFID technologies," *European Conference on eHealth 2006, Proceedings of the ECEH 2006*, pp. 131–142, 2006.

[15] EPCglobal, "Specification for RFID Air Interface EPC ™ Radio-Frequency Identity Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860 MHz – 960 MHz," *Intellectual Property*, no. October, 2006.

[16] A. Dementyev, S. Hodges, S. Taylor, and J. Smith, "Power Consumption Analysis of Bluetooth Low Energy, ZigBee, and ANT Sensor Nodes in a Cyclic Sleep Scenario," in *Proceedings of IEEE International Wireless Symposium (IWS)*, IEEE, 2013.

[17] J. N. S. Rubí and P. R. L. Gondim, "IoMT platform for pervasive healthcare data aggregation, processing, and sharing based on oneM2M and openEHR," *Sensors (Switzerland)*, vol. 19, no. 19, pp. 1–25, 2019.

[18] RedHat, "Cloud Computing - IaaS vs PaaS vs SaaS."

[19] A. F. Subahi, "Edge-Based IoT Medical Record System: Requirements, Recommendations and Conceptual Design," *IEEE Access*, vol. 7, pp. 94150–94159, 2019.

[20] B. Xu, L. D. Xu, H. Cai, C. Xie, J. Hu, and F. Bu, "Ubiquitous data accessing method in iot-based information system for emergency medical services," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1578–1586, 2014.

[21] IBM, “Application Programming Interface (API).”

[22] HL7, “FHIR v4.0.1,” 2019.

[23] C. Peng and P. Goswami, “Meaningful integration of data from heterogeneous health services and home environment based on ontology,” *Sensors (Switzerland)*, vol. 19, no. 8, 2019.

[24] J. Gruendner, T. Schwachhofer, P. Sippl, N. Wolf, M. Erpenbeck, C. Gulden, L. A. Kapsner, J. Zierk, S. Mate, M. Stürzl, R. Croner, H. U. Prokosch, and D. Toddenroth, “Ketos: Clinical decision support and machine learning as a service – A training and deployment platform based on Docker, OMOP-CDM, and FHIR Web Services,” *PLoS ONE*, vol. 14, no. 10, pp. 1–16, 2019.

[25] K. B. Wagholikar, J. C. Mandel, J. G. Klann, N. Wattanasin, M. Mendis, C. G. Chute, K. D. Mandl, and S. N. Murphy, “SMART-on-FHIR implemented over i2b2,” *Journal of the American Medical Informatics Association : JAMIA*, vol. 24, no. 2, pp. 398–402, 2017.

[26] BrainAnswer, “e-CoVig - Automated COVID-19 Symptomatology Remote Monitoring System.”

[27] F. Dursun Ergezen and E. Kol, “Nurses’ responses to monitor alarms in an intensive care unit: An observational study,” *Intensive and Critical Care Nursing*, vol. 59, no. xxxx, p. 102845, 2020.

[28] Personal Connected Health Alliance, “Continua Adoption Playbook Deploying Interoperable Connected Health in Your Health System,” no. May, p. 20, 2017.