20th IEEE International Conference on E-Health Networking, Application and Services (HealthCom), Ostrava, Czech Republic, September 2018.

Brokering Services for Integrating Health Cloud Platforms for Remote Patient Monitoring

Raafat Aburukba, Assim Sagahyroon, Fadi Aloul, Niha Thodika
Computer Science and Engineering
American University of Sharjah
Sharjah, United Arab Emirates
raburukba@aus.edu, asagahyroon@aus.edu, faloul@aus.edu, g00057302@aus.edu

***Abstract*— Remote patient monitoring (RPM) uses technology to assess patients' health within and beyond the healthcare facilities. RPM is becoming a reality by connecting the patient's healthcare devices to the Internet. Modern healthcare devices come with a cloud-based support that provides patients with new features such as the broad network access to their vitals' readings from anywhere. This shift of cloud-based support brought a new way to interact with the patients' devices, as well as, new challenges that are different from the traditional Bluetooth interaction that is heavily studied in the literature. In real situations, patients usually have multiple devices to measure different vital signs that are made by different providers. Hence, it is not possible with the current approaches to have all patient vital readings consolidated in a single location. This consolidation enables the management of the chronic disorders and to analyze the health trends as part of the valuation of the treatment process. This work proposes a cloud-based brokering solution that seamlessly integrates multiple health cloud providers through a set of services that uses the web services cloud standard for the purpose of remote patient monitoring application. This work is validated by a prototype implementation.**

Keywords— Remote Patient Monitoring; Cloud Computing; Healthcare; Integration Services; Health Brokering Services

I. INTRODUCTION

Several studies indicated the need for approaches to minimize the cost of patient healthcare while maximizing the quality of health services [1]. There are growing trends in the number of chronic diseases such as hypertension, diabetes, asthma, cardiovascular problems, etc. Consequently, the costs of healthcare delivery is increasing and access to healthcare facilities in rural areas is limited.

Remote patient monitoring (RPM) is the application of technology to assess patients' health within and beyond the healthcare facilities. There are two major types of RPM: 1) long-term which deals with the management of chronic disorders, and analyses the health trends that are monitored as part of the evaluation of the treatment process; 2) real-time emergency detection and alerts of patients with vital signs. The main purpose of RPM is to:

- Reduce the need for checkups and hospitalization
- Continuous assessment of the treatment effectiveness
- Extends independent living for elderly
- Allows caregivers to provide services for greater number of patients

The quality of healthcare can be based on the suitable treatment that is provided to patients at the right time. Hence, RPM contributes prominently in the improvement of the healthcare quality. This can be done by making health predications, accurate advice, and healthcare planning based on defined intelligence.

In an eHealth patient survey [2], it was inferred that 33% of patients want their physicians to have access to RPM technologies and 40% of older patients want to alert their caregivers when they have a health emergency. Thus, the use of RPM technology can help provide a viable solution. This explains the exponential increase in RPM technologies as supported by the article which states that there is a 44% growth in the RPM market in 2016 [3].

Given the heterogeneous nature of the patient sensors, integrating the different protocols and data format that might be encrypted is a challenge. Furthermore, patients have different health conditions and treatment procedures based on multiple parameters such as gender, height, age, etc.

Modern healthcare devices come with a cloud-based support that provides patients with new features such as the broad network access to their vitals' readings from anywhere. Cloud computing paradigm delivers compute, platform, and software as services.

Cloud computing brought many advantages to different domains. In healthcare, cloud computing provided the advantage of the high availability of patient data and the ability to access it from anywhere at any time. On the other hand, the existence of multiple cloud providers where different patient data may reside without having the ability to interoperate is still an open issue. Some experts predict that cloud computing can make healthcare cost effective and efficient [4]. However, this is not possible with the current trends without enabling the interoperability between multiple cloud healthcare providers. The work in [5]-[9] show the surge in research activities that advocate cloud computing in the healthcare domain. The benefits of cloud computing in health was summarized in [10] and [11] as follows:

- Access to computing and storage resources on-demand bases.
- Relieves the need of the technical skills in building the physical infrastructure and management, as well as from the data backups, and recovery from the healthcare domain.
- Provides business intelligence capability and data visualization ability.

Hence, cloud computing can reduce the overall cost of healthcare, permits increased access to critical data from different geographical locations and by different stakeholder, and provide a much needed scalability and elasticity to the demand for health services.

Although there are many health devices that checks for patient's vital signs, given the diverse nature of manufacturers, it is difficult for the patients to find a common platform for all of their vitals monitoring devices. Nowadays, it is common that each health monitoring device has their proprietary application and cloud platform in which the patient vital data is transferred and stored. Consequently, the main objective of this work is to develop a unified platform that is capable of collecting the data from the patient's devices through the vendor specific cloud healthcare platform. The system can react according to defined rules by the doctor. These rules are formed depending upon the patient's profile and health condition. The patient vital data generated by the patients' devices can be viewed by the patient and the doctor simultaneously, whereas doctors can make more accurate diagnosis and treatment decisions based on the obtained readings.

This paper is organized as follows, Section 2 presents existing approaches, Section 3 presents the environment analysis and requirements, Section 4 presents the proposed architecture deployed in the cloud platform as a service, and Section 5 presents the implementation, and Section 6 presents the conclusion remarks of this work.

II. Literature Review

The work in the literature captured the data from patient devices through a gateway with the assumption that all connected devices utilize the same communication standard. The authors in [12] presented a Service Oriented Architecture (SOA) conceptual platform for m-Healthcare. The emphasis was on the SOA as an enabler technology for integrating devices with the computational servers. Another work in [13] presented a mobile cloud infrastructure for assistive healthcare. Cloud computing was used for storage and data sharing. Mobile agent were also deployed in the gateway device to collect the patient data from the devices. iHealth [14] on the other hand provide a commercial healthcare cloud platform. It enables iHealth device integration. Current iHealth devices utilize two communication standards, WiFi and Bluetooth. The platform also supports the integration of those standards. The devices Bluetooth capability transmit the data to the iHealth cloud via the mobile device as a gateway. The devices with the WiFi capability transmit the data to the iHealth cloud directly. The iHealth platform only integrates iHealth based devices. However, it provides a set of web-service APIs to collect the data from the iHealth patient devices.

Similarly, Nokia [15], Fitbit [16], and MySignals [17] have a cloud platform that enables the collection and access of patients' records through their own cloud platforms. They also provide access to their APIs via web-services. Hence, there is a trend of modern healthcare devices being manufactured with the health cloud platforms that support data collection and utilize the web-service standard in exposing APIs for possible integration with the healthcare platforms.

None of the current work present cloud-based services that enable the seamless integration of multiple cloud healthcare vendors to consolidate the patients' device readings in one platform. The aim of this work is to propose a healthcare brokering services that integrates multiple health cloud platforms by using the web-services standard.

III. Environment Overview and Requirements

In RPM, a patient may have m devices denoted by $D = \{D_1, D_2, \ldots, D_m\}$. D_i is a physical device that is defined by a set of capabilities $D_i = \left\{c_i^1, c_i^2, \ldots, c_i^{d_j}\right\}$, where d_j is the number of capabilities that describe D_i. In this work, we assume that each device is linked to a cloud vendor specific platform $V_k \in V$. We denote the cloud vendors by $V = \{V_1, V_2, \ldots, V_z\}$. Each cloud vendor expose specific services with specific capbilities denoted by $V_k = \left\{s_k^1, s_k^2, \ldots, s_k^{v_l}\right\}$, where v_l is the number of services provided by cloud vendor V_k.

To monitor the patient's health based on their condition, we define a set of n activities denoted by $A = \{A_1, A_2, \ldots, A_n\}$ where activity $A_j \in A$ $(j = 1, \ldots, n)$ is formulated as a set of rules $A_j = \left\{r_j^1, r_j^2, \ldots, r_j^{t_j}\right\}$ where t_j is the number of rules belonging to A_j . For example, activity 1 can have the following rule:

if the blood sugar average value in the last 10 days is over 15 then: send an alert to the doctor

Caregivers can assign multiple rules to a patient depending on the patient's health condition. Moreover, a rule may have a dependency relationship with other rules. Depending on the defined rules and the current patient condition, the decision is to allocate the right resources (such as specialized the hospital with adequate resources, and the doctor with the required specialization), to the patient, at the right time.

Figure 1 presents a high-level view of the proposed work. The figure shows patients with specific sensors to monitor their vitals. The sensor devices are integrated with its vendor cloud platform. The proposed healthcare brokering services integrate the device specific health cloud providers and collects the data generated by the devices through the components specified in section IV. The healthcare brokering services are deployed in a cloud Platform as a Service (PaaS) in which the PaaS is deployed on a cloud Infrastructure as a Service (IaaS). Security and privacy are critical as communication of individual patient data are transmitted over cyber-physical networks. This layer is presented across all the other layers within the presented architecture in Figure 1. This means security and privacy must be considered in all layers within the overall architecture.

In distributed RPM applications, sensory data collected by the various gateways eventually needs to be transferred to the cloud for long term storage. Electronic Medical Records (EMRs) in healthcare refers to the storage of all healthcare data and information in electronic formats with the associated information processing and knowledge support tools necessary for the managing the health enterprise system [18].

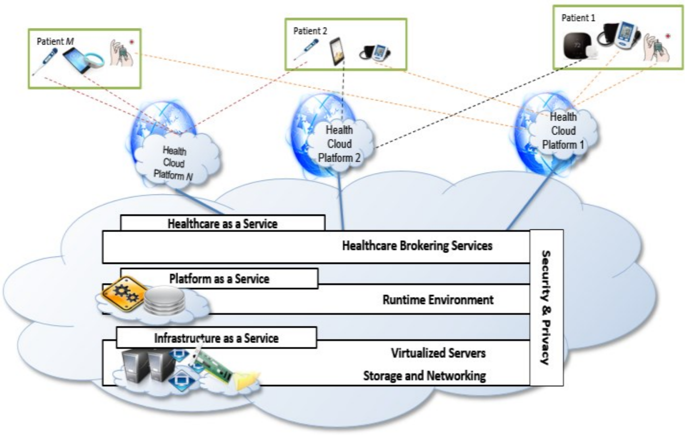

Fig. 1. Proposed Conceptual View

IV. Proposed Healthcare Brokering Services

This work presents a healthcare brokering services that enables the integration of multiple healthcare cloud platforms. In addition, this framework enables patient's health monitoring, analysis, and coordination between devices and other systems. The framework is deployed on a cloud platform as a service (PaaS). Security and privacy are critical as individual patient data is communicated over the cyber-physical networks. This layer is presented across all the other layers within the presented architecture in Figure 1. This means security and privacy must be considered in all layers within the overall architecture.

The proposed healthcare Brokering services are presented in using the Unified Modeling Language (UML) component diagram in Figure 2. Those components reside in the healthcare brokering services layer presented in Figure 1. In the component diagram presented in Figure 2, the shaded components are external components that present the vendor specific cloud platform that our proposed services integrate with. Details about the components are provided in the following subsections.

A. *Caregiver User Interface Componenet*

This component provides a webportal to the caregiver to interact with the system. The features provided through this interface are provided for their respected role in the system.

B. *Patient User Interface Componenet*

This component provides the ability for the patient to interact with the system, and the system to interact with the patient via a mobile based interface. The features are specifically provided for their role as a patient.

C. *User Management:*

This component deals with the creation and management of user accounts It delivers the following main functionalities:

- Manage accounts: Handles the functionalities to create, update, and delete user accounts.
- Associate caregiver to patient: gives the ability to associate multiple caregivers (specialized doctors) to patients.
- Associate user account to patient: gives the ability to associate registered accounts in the platform to the patient.

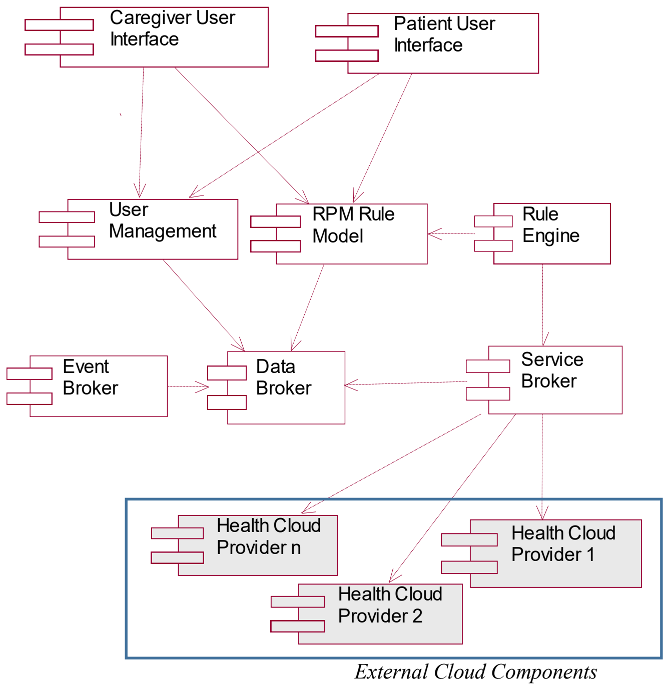

Fig. 2. Healthcare Brokering Services UML Component Diagram

D. *Event Broker*

This component allows events generated by specific devices to be registered and published. Moreover, services can subscribe to the published events and as those events are triggered, the subscribers are notified. For example, readings from a specific patient's device can be registered with the event broker. As an event from the patient's device is generated, any subscriber to this event will be notified. Hence, the event broker component has the following main modules:

- Event registration: integrated devices or services can produce events. The service that produces the event must be published under a specific topic. This module allows the creation of the event topic, the identification of service or device that generates the event, and the option to make this event public or private. If the event registered is set to public, it will allow other services to subscribe for this topic. On the other hand, if the event registered is set to private, it will have limited access to specific services to access the registered topic.
- Retrieve event definition: provides the event listings and an interface definition of all registered events.
- Event monitor: monitors the occurrence of an event on a patient's device or as exposed by other services in the system.
- Event trigger: executes the set of actions when an event is triggered.
- Event coordinator: deals with the coordination between the event producer and consumer when an event is triggered to execute the defined set of actions.
- Link event to service: links specific event to a defined service. This allows to link the healthcare vendor cloud service to an event.

E. Service Broker

This component enables the integration of multiple cloud provider services. It also deals with the specific way to authenticate and authorize access to the patient's data that are captured from their devices. The service broker has the following main modules:

- Service definition: allows the ability to describe the health cloud provider exposed services, its attributes, and the expected return from the service.
- Service publication and discovery: provides the ability for the defined services to be discovered and used based on the healthcare devices used by the patient.
- Service selection: provides the ability to select the right service to interact with based on the devices owned by the patient.
- Associate service to service: this module gives the flexibility to associate multiple services under a specified group where each service is given a specific priority. This enables the flexibility to define dependency relationships between defined services. For example, to read the data of Patient x, that owns device y, provided by vendor z, the patient and application must be authenticated and authorized respectively to access the patient's health data.
- Invoke service: connects and invokes to the defined service on the cloud health provider.

F. Data Broker

This component decouples the data storage and data format from the core logic of the system. This gives the flexibility to integrate multiple patient devices and health cloud services without changing the logic of the system. The following modules enable data persistence and retrieval of the patient's data intelligence, as well as the cloud healthcare providers' service definitions.

- Data access: access the patient data. The module provides the retrieve, insert, delete, and update operations to the patient's data.
- Data extraction: extracts the data received from the cloud healthcare provider based on the defined and registered data types.
- Data transformation: deals with any transformation to the data to match the destination acceptable format.
- Data loading: this module loads the data depending on the request, if it is a request for basic data, then the basic data is loaded. On the other hand, if the requested data has references to other data, then the basic data is loaded along with the setup to the references of the other referenced data.
- Data monitoring: this module monitors specific patient data changes. It is consumed by the event broker component to monitor data changes related to a patient's sensor device through the defined healthcare cloud platform provider.

G. RPM Rule Model

The healthcare model enables the definition and management of rules for customizing patient's condition, notifications, and alerts. The model is also used by the rule engine. The Healthcare model contains three main sub-modules:

- Meta-model: this includes the structure to define rules.
- Model instance: uses the meta-model to enable the healthcare giver to define rules for a patient.
- Model management system: enables the healthcare giver to manage (add, update, delete) rules. The healthcare giver can create templates of models that can be used for specific class of patients that may share the same conditions

The rules model describes a statement that defines or constraints some aspect within the healthcare domain. A rule in healthcare is intended to control or influence the doctor or patient behavior. A rule engine evaluates and executes rules, which can be expressed as if-then statements. The power of healthcare rules lies in their ability to separate the knowledge from its implementation logic and to be changed without changing the implementation. A rule is composed of two parts, a condition and an action. When the condition is met, the action is executed [19][20].

The concept of rules captures precise logic in healthcare, requests, and time to govern the behavior of the decision module. Five RPM rule types are presented:

- Patient Rules: are related to the patient health condition.
- Time Rules: are related to time such as morning time.
- Resource Rules: are related to the resources within the healthcare domain. Those resources can be physical devices, human resources, or specific health facility.
- Patient-Time Rule: associates defined patient rules to specific defined time rule. This association for example can define rules to specific values by the patient's devices at a specific time period.
- Resource-Time Rule: associates defined resource rules with the defined time rules. This association allows to set specific resource availability during a specific time.

H. Rule Engine

The rules are evaluated for following main purposes provided by the decision module:

- Allocating patients to resources at a specific time based on the defined rules.
- Notifications and alerts are triggered depending on the defined rules.

The decision is made based on the patient rules, resource rules, time rules, patient-time rules association, and resource-time rules association.

V. Implementation

This work was validated by building a prototype with the components mentioned in the previous section. The implementation made use of Java as the programming language and used RESTful web-services to integrate with the external health cloud providers [14][15][16]. The proposed brokering services were deployed on a cloud PaaS provider [21]. Figure 3 shows the implementation high-level view which integrates two patients, and six different devices integrated by the three different providers. The devices used in the prototype implementation are: glucometer by iHealth, a blood pressure monitor by two different vendors namely iHealth and Nokia, Fitbit watch, and pulse oximeter by iHealth. Moreover, we integrated an Android based smart watch via Bluetooth with the mobile device. The mobile application acted as a gateway to capture the smart watch device readings and integrate those readings with the proposed brokering services by calling the service broker component. All service definitions of the device manufacturer cloud provider are defined within the service broker component. The service broker component invokes the proper services based on the incoming requests.

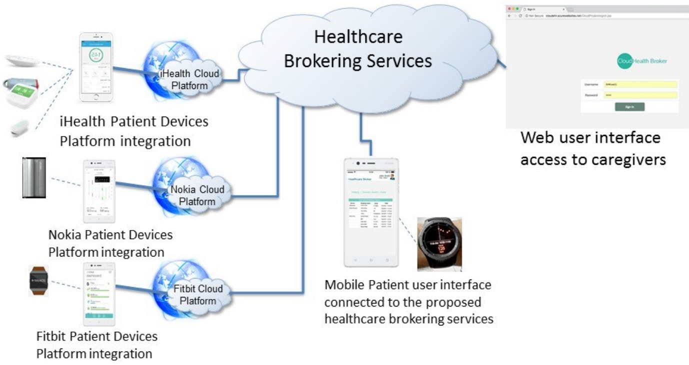

Fig.3. Implementation High-level view

In order to retrieve the patient's data from the cloud health providers, the patient must be authenticated. The cloud health providers utilize the OAuth standard protocol [22]. Once a token is received in JASON format, the proposed healthcare brokering service access the registered devices data through the cloud health provider's platform.

The patient's data is accessed through a request triggered by an authorized user (such as the patient, caregiver, or any associated user to the patient), in which the service broker component connects with the healthcare platform provider and captures the data read from the devices.

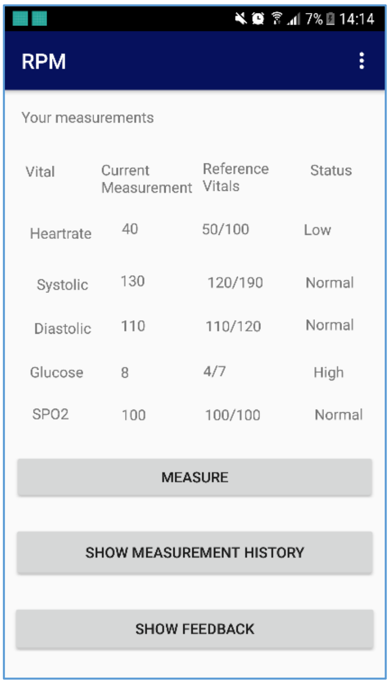

Fig. 4. Prototype Screenshot of the Mobile Application

The service broker component also subscribes for the device manufacturer cloud provider to receive the data as they are captured from the patient's devices. Those data are integrated and stored by calling the modules provided by the data broker component. The data is used by the RPM model and it triggers the specific actions as defined by the rules. If some of the rules require healthcare resources (such as the doctor and hospital), the registered resources are notified with the location of the patient.

Figure 4 shows the prototype screenshot of the patient's mobile application. The figure shows the device readings that were integrated through the vendor specific cloud platform. The status presented in Figure 4 shows a feedback status of the reading based on defined rules by the caregiver. Those rules are transferred to the patient's mobile device and are compared against the actual readings. Figure 5 shows the mobile notifications based on the defined patient rules. The patient rules instance is transferred to the patient's mobile device and the rules are monitored and the notification is triggered.

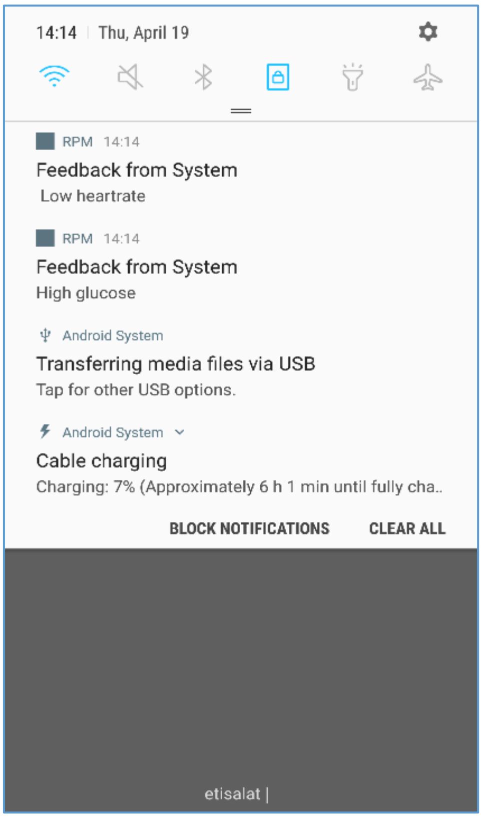

Fig. 5. Notification to the patient on their mobile device

VI. CONCLUSION

This work proposed a brokering framework model for RPM that integrates multiple cloud health providers and incorporates customized rules for each patient. A decision making system is developed to allocate the proper resources to the patient at the right time. This work was validated by creating a prototype implementation that integrates five devices that are connected through three different cloud health platforms. The prototype presented the ease of integration a cloud health platform and capturing the readings of the patient's devices.

Acknowledgment

The authors would like to thank Brinda Karumathil Jayadevan and Naili Mustafa for their help in the implementation. The authors would like to also thank the Computer Science and Engineering department at the American University of Sharjah, UAE, for providing the support and resources. This research was partially funded by the American University of Sharjah FRG grant number FRG17-R-20.

References

[1] S. Hochron, P. Goldberg, "Driving physician adoption of mheath solutions," Healthc. Financ. Manage. 69(2), pp. 36-40, February 2015.

[2] B. Ward, J. Schiller, R. Goodman, "Multiple chronic conditions among US adults: a 2012 update. Prev Chronic Dis," 2014.

[3] J. Morrissey, "Remote patient monitoring: How mobile devices will curb chronic conditions," in Factor in the time and cost of implementing data-driven technology, ed: UBM Medica, 2014.

[4] A. Kuo, "Opportunities and challenges of cloud computing to improve health care services," Journal of Medical Internet Research, 13(3), 2011.

[5] S. Rallapalli, R. Gondkar, U. Ketavarapu, "Impact of Processing and Analyzing Healthcare Big Data on Cloud Computing Environment by Implementing Hadoop Cluster," Procedia Computer Science, Vol. 85, pp. 16-22, 2016.

[6] W. Wang, L. Chen, Q. Zhang, "Outsourcing high-dimensional healthcare data to cloud with personalized privacy preservation," Computer Networks, Volume 88, pp. 136-148, September 2015.

[7] R. Sanz-Requena, A. Mañas-García, J. Cabrera-Ayala, "Cloud-Based Radiological Portal for the Patients: IT Contributing to Position the Patient as the Central Axis of the 21st Century Healthcare Cycles," IEEE/ACM First Intl. Workshop on Technical and Legal Aspects of Data Privacy and Security, May 2015.

[8] G. Bhatia, A. Lala, A. Chaurasia, R. Rajpal, "Implementation of cloud computing technology for the improvement of the entire healthcare services in India," International Conference on Advances in Technology and Engineering, 2013.

[9] W. Yu, M. Kollipara, R. Penmetsa, and S. Elliadka, "A distributed storage solution for cloud based e-Healthcare Information System," IEEE 15th International Conference on e-Health Networking, Applications and Services (Healthcom), October 2013.

[10] "Advancing Healthcare Delivery with Cloud Computing," a report by The European Coordination Committee of the Radiological, Electromedical and Healthcare IT Industry. Available at: http://www.cocir.org/fileadmin/4.4__eHealth/eHealth_Toolkit_INT_2012_chap2.pdf.

[11] "Impact of Cloud Computing on Healthcare," a report by the Cloud Standards Customer Council. Available at: http://www.cloud-council.org/deliverables/CSCC-Impact-of-Cloud-Computing-on-Healthcare.pdf.

[12] W. S. Ng, J. C. Ming Teo, W. Tiong Ang, S. Viswanathan , and C. Khong Tham, "Experiences on developing SOA based mobile healthcare services," IEEE Conference in Services Computing, pp. 498-501, December 2009.

[13] D. B. Hoang, L. Chen, "Mobile Cloud for Assistive Healthcare (MoCAsH)," IEEE Conference in Services Computing Conference (APSCC), pp. 325-332, December 2010.

[14] iHealth Cloud Services. Available at: https://cloud.ihealthlabs.com

[15] Nokia Health Cloud Services. Available at: https://health.nokia.com

[16] Fitbit Cloud Services. Available at: https://dev.fitbit.com

[17] "MySignals-eHealth and medical IOT Development Platform". Available at: http://www.my-signals.com/

[18] T.J. Hannan, "Electronic medical records. Health informatics: an overview," Churchill Livingstone, Australia, pp. 133-148, 1996.

[19] C. Leong, A.R.Ramli, and T. Perumal, "A Rule-Based Framework for Heterogeneous Subsystems Management in Smart Home Environment," IEEE Transactions on Consumer Electronics, vol. 55, no. 3, pp. 1208-1213, August 2009.

[20] S. M. Huynh, D. Parry, A. C. M. Fong, and J. Tang, "Novel RFID and Ontology based Home Localization System for Misplaced Object", IEEE Transactions on Consumer Electronics, vol. 60, no. 3, pp. 402-410, August 2014.

[21] Amazon AWS. Available at: https://aws.amazon.com.

[22] iHealth Authentication. Available at: http://developer.ihealthlabs.com