International Journal of Smart Home
Vol. 10, No. 9 (2016), pp. 329-346
http://dx.doi.org/10.14257/ijsh.2016.10.9.31

PDConnect: Low Cost Wearable Device connecting Patient-Doctor via Cloud for Good Health

Akarsh Goyal, Ishan Khandelwal, Aditya Kumar Mathur,
Ronnie D. Caytiles* and N.Ch. S.N. Iyengar

SCOPE, Vellore Institute of Technology University, Vellore-632014, TN, India
**Department of Multimedia Engineering, Hannam University, Korea,*
akarsh.goyal15@gmail.com, ishankhandelwal23@gmail.com,
adityamathur09@yahoo.com,
rdcaytiles@gmail.com, nchsniyr@vit.ac.in

Abstract

Cost effective and time saving solutions are available for all of our health concerns due to the rapid growth of Technology. But the rural areas of many countries continue to suffer due to the lack of good health services with doctors. Our model focuses on connecting the rural villages to quality doctors via cloud computing and wearable technology. The system could help to monitor and manage good health with doctor's interaction in emergence. It's a wearable jacket where health measuring sensors will be embedded. It sends all the vital stats in real time to the cloud where the authorised doctors can access the information after user's request. The records contain the patient's medical history. It's a user-friendly application holding the whole database of the patients securely in the cloud with real time updates. Patients can refer to the available Doctor anytime for treatments and diagnosis. The system alerts to the nearest doctor in risk case. This system if is put in use at the rural areas then there will be much benefit to the rural people in their health concerns. They can lead a healthier and more independent lifes.

Keywords*: smart devices, doctor-patient consultation, cloud, wearable, sensors, rural areas*

1. Introduction

Healthcare is a very important aspect to ensure a healthy body, a healthy workplace, a healthy community and a healthy nation. The individuals, the workplace our National security are threatened when nation suffers from a lack of good health. Thus researchers try to design innovative methods for the health of the country. A small component of medical research effort can improve the quality of life and nation. Rural areas are still suffering from better healthy life due to the lack of facilities and experienced doctors.

So to reduce this disparity related to health between cities and villages we have come up with the idea of Patient-Doctor Connect. via body area network [4][12] with low cost sensors [16] affordable by a common man. The readings communicated by the sensors lively will be stored in the patient database in the cloud, can be viewed by both the patient and the registered doctor via smart mobile. The protocol used for this will be MQTT which will be brokered on a cloud platform.

In case of emergency an alert containing the details of the patient will be sent to the nearest doctor via the application. The life of the patient can be saved on proper emergency action. This integration of a wearable device and a mobile health application could be better useful thing in the healthcare industry. It may revolutionize the healthcare system in rural areas.

ISSN: 1975-4094 IJSH

2. Related Work

Studies show that storing the data in web database is a viable option for implementing an Android based patient monitoring system for hospitals. An Android based patient monitoring system is portable, compact, mobile and makes storing data in database very simple [7]. A medical application can help the patient by saving the time of travel to visit a clinic and the doctors can treat the patients from their places at a time convenient to them. The application is user-friendly and easy to install in a smartphone [3].

ICT (Information and Communication Technology) devices such as smartphones playing major roles in various industries including the Health Care Industry [1]. Since many years the Health Care Industry has used the traditional paper based system for data management and appointment scheduling. Prema Sundaram [11] has developed an Agent Based Hospital Appointment Scheduling System that uses the mobile agents to schedule appointments based on fuzzy preferences.

A system was proposed by [14], which consists of two panels: Doctor and the Patient. The patient registers into the application and creates a unique profile which can be accessed via a username and password. After viewing the doctor profile the patient can request an appointment with a doctor at the desired time.

In [9] implementation of wireless sensor networks in association with Arduino has been discussed. The entire system architecture and hardware part has been provided. [4] as well as [12] focus on making a body area network for monitoring the health of the patient.

In [16] the authors presented the u-healthcare system in the Internet of Things (IoT) environment with the support of the mobile gateway which makes it possible for integration. In [6] the authors developed a smart elderly home monitoring system (SEHMS).

In [2] the author suggests a binding protocol which enables users to use public medical devices as if they are their own medical devices through IoT Healthcare. In [8] the author proposes a handheld Compute Engine (CE) for Personal Health Devices (PHDs). The CE is a device that receives measurements from more than one PHD, and collects, analyses and displays the received measurements.

In [10] the author's approach is based on time series analysis that will allow the computer to determine whether a stream of real-time sensor data contains any abnormal heartbeat.

3. Novelty of the Idea Proposed

Comparisons have been made with other similar systems which are currently there in the market. We see that our model is distinct from the other pre-existing technologies on the basis of the attributes mentioned below. So the application could be made more superior in achieving its objective as compared to its predecessors by proper refinement.

Table 1. Comparison with Pre-Existing Ideas

Name	Company Name	Product Status	Multi Sensor	Cloud Storage	Co ordinates Mapping	Automatic User-Doctor Connect	Technology compatibility
Intelligent Asthma Management [21]	Health care Originals	Production phase	Only Diabetes	No	No	No	-
Vitaliti [18]	Cloud DX	Patented	Yes	Yes	No	No	Bluetooth, IOS, Android
Abbott Diabetes Care [17]	Abbott Diabetes Care	Production/ Licensing Phase	Only Diabetes	Yes	No	No	Smartphone accessible
Embrace [19]	Embrace	Already in market	Yes	Yes	No	No	Bluetooth, IOS, Android
Qardiocore [20]	Qardio	Production/ Licensing Phase	Yes	Yes	No	Yes	Bluetooth, IOS, Android
Valedo Back Therapy Kit [24]	Hocoma	Already in market	Yes	Yes	No	No	IOS, Android, WI-FI, Bluetooth
Helius [22]	Proteus Digital Health Inc.	Already in market	Yes	Yes	No	Yes	IOS, Android, WI-FI
Quell Relief [23]	NeuroMetrix, Inc.	Already in market	Yes	Yes	No	No	Smartphone accessible, Bluetooth
PDConnect	-	Prototype	Yes	Yes	Yes	Yes	IOS, Android, WI-FI, GPS

4. Proposed Model with Working Procedure

The wearable comprising of various sensors [9] is attached to the patient's body via a jacket as shown in the figure above. The sensors are connected to the Arduino board which also has a Wi-Fi Shield so as to connect it with the Wi-Fi [14]. These sensors will be giving live readings of the body attributes they are monitoring to the application through the MQTT protocol. The protocol resides in the cloud to which readings are published. From here the patient could subscribe so that he receives the readings in his smart device [15]. These will get saved in the patient database. Also the patient could provide his previous medical history to be stored in the database. A doctor chosen by the patient can monitor his health by going through his records and the readings. Benchmark values for different parameters are set properly in the cloud on the basis of which the patient's health is being monitored. So if any of the attributes cross the threshold then it will be directly communicated to the nearest doctor. The alert will be giving the unique id and the GPS coordinates of the patient. This is done by an application running in the background. The doctor could then take swift action like calling a health centre or an ambulance service as considered appropriate by him.

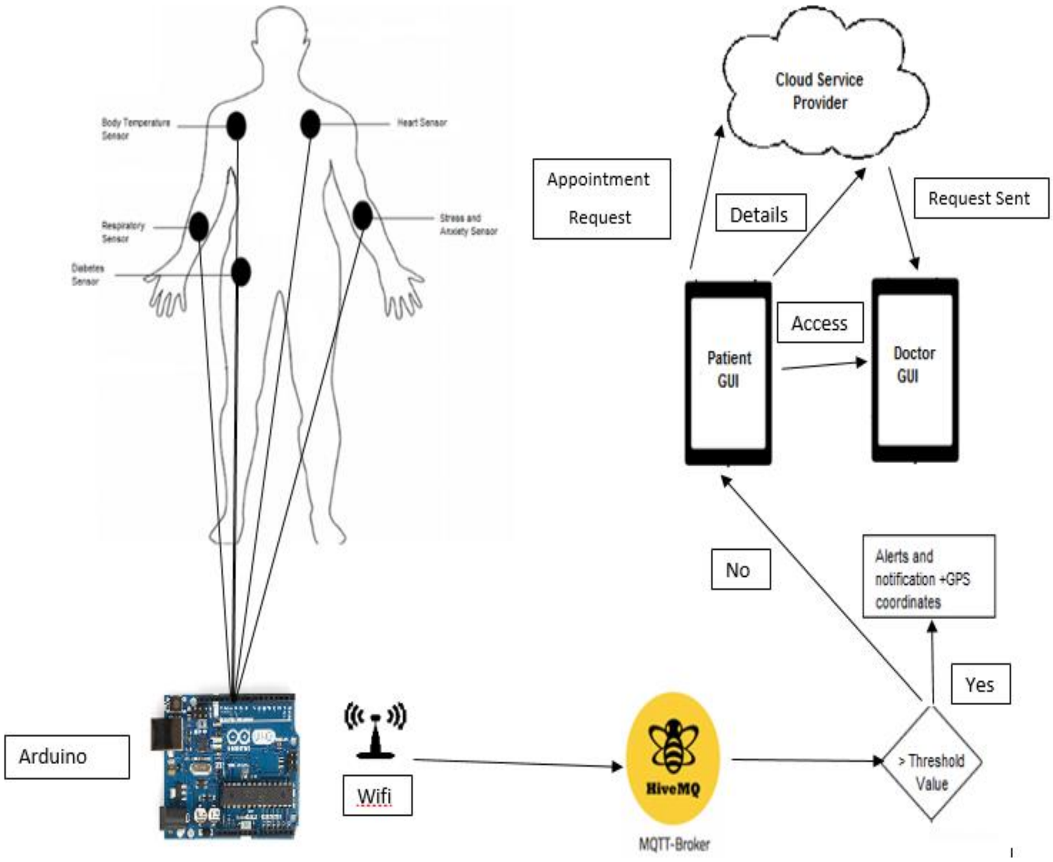

Figure 1. Entire Model

5. Technical Requirements

5.1. Hardware Specifications

a. Sensors – The sensors used in PDConnect are–

i. Heart Rate Sensor - The heart rate sensor is based on the principle of photo plethysmography. If there is a change in light intensity in any organ caused by change in volume of blood then it will be detected by the sensor. It usually has an operating current of 100mA at +5V dc regulated. Also it could be used to detect blood pressure as well at a given time.

ii. Respiratory Sensor - Breathing monitoring is one of the most important elements in assessment of physiological state. It can provide valuable information related to cardiac, neurological and pulmonary conditions. The respiratory sensor has a bandwidth of 0-15 Hz and consumption of 4mA.

iii. Stress and Anxiety Level Sensor - Its principal task is to detect whether a person is stressed at a particular moment in time or not. It is done by measuring galvanic skin response (GSR). GSR is affected by the amount of sweat on the different body parts. We will be using one stress and anxiety level sensor with consumption of 5mA at +5V.

iv. Body Temperature Sensor - This sensor allows to measure body temperature. It is of great medical importance to measure body temperature. The reason is that a number of diseases are accompanied by characteristic changes in body temperature. Likewise, the course of certain diseases can be monitored by measuring body temperature. So a body temperature sensor with consumption of 4mA at +5V will be used.

v. Diabetes Sensor –This sensor records glucose level continuously around the clock. It allows you to see how fast and in what direction glucose level is trending. It has a bandwidth of 0-25Hz and consumption of 6mA. The signal represents the amount of sugar in the blood.

These sensors will be integrated together to form the body area network.

b. Arduino – The sensors will be connected to Arduino Uno. It will have 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It will contain everything needed to support the microcontroller. We will simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

c. Wi-Fi Shield – The Arduino Wi-Fi Shield will allow the Arduino board to connect to the internet using the 802.11 wireless specification (Wi-Fi). It is based on the HDG204 Wireless LAN 802.11b/g System in-Package. An AT32UC3 provides a network (IP) stack capable of both TCP and UDP. We will use the Wi-Fi to write sketches which connect to the internet using the shield. There is an on-board micro-SD card slot, which can be used to store files for serving over the network. It is compatible with the Arduino Uno. The on-board microSD card reader is accessible through the SD Library.

5.2. Software Specifications

a. MQTT Protocol Setup – MQTT stands for MQ Telemetry Transport. It is an extremely simple and lightweight messaging protocol, designed for constrained devices and low-bandwidth, high-latency or unreliable networks. The design principles are to minimise network bandwidth and device resource requirements whilst also attempting to ensure reliability and some degree of assurance of delivery. These principles also turn out to make the protocol ideal of the emerging “machine-to-machine” (M2M) or “Internet of Things” world of connected devices, and for mobile applications where bandwidth and battery power are at a premium. Thus the MQTT protocol is used as the standard for getting the readings from the sensors to the mobile application. Network security and privacy are also ensured by using it. The user-name and password are passed with a MQTT packet in V3.1 of the protocol. Encryption across the network can be handled with SSL, independently of the MQTT protocol itself. So the MQTT protocol will reside in the cloud service where the data from the sensors is sent and the patients will receive this data when they subscribe to the Cloud MQTT.

b. Application Development – The application is developed on Android as it has a wide reach. The software to be used for application development is Android Studio. It is an integrated development environment (IDE) for developing for the Android platform. Based on Jet Brains IntelliJ IDEA software, Android

Studio is designed specifically for Android development. The application is lightweight and easily accessible to the doctor and patients. The data sent by the sensors is stored using SQLite in the app.

c. Cloud Storage - Cloud storage is the backbone of the project here as it stores all the data which includes all the data that the patient enters while signing up. Storage plays a pivotal role as it provides almost unlimited data storage capacity while ensuring the all the important data is backed up. It also makes backup and data recovery much simpler than the traditional methods used for data storage. Also the cloud service is used as a broker to implement the MQTT protocol where the data is published by the wearable part. This data is sent to the user who subscribes to it. There are many Cloud Service Providers available who help in monitoring and managing the cloud based applications by providing excellent analytical support and push services to create a direct channel of communication between the user and the cloud storage. HiveMQ cloud service is used for the purpose of cloud storage as it provides proper support for data analytics and also a lot of free storage is provided to the users.

d. Online Video Chat Services - Online video-chat services empower the doctors to provide e-consultations to patients residing in far off cities. Video appointments let doctors check in quickly with patients, which can be more convenient for both. APIs such as Sinch, WebRTC or TokBox can be used to integrate video chat service in the android app.

6. System Design

The overall steps are as summarised by the following sub-sections –

6.1 Wearables

The wearable will measure heart rate, body temperature, blood sugar levels, respiratory rate, stress and anxiety levels. The sensor nodes contains sensors and microcontroller [16] (in this case Arduino Uno). Sensors acquire the data and send it to microcontroller. All these sensor send the analog data

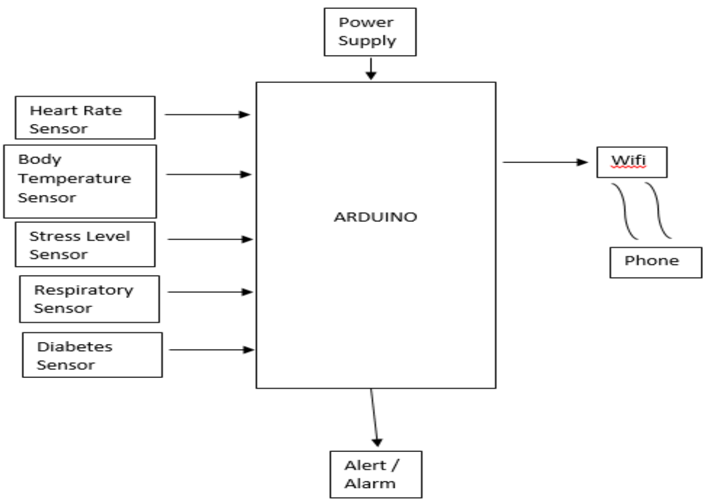

Figure 2. Arduino Connections

to micro-controller. To make the data understandable to microcontroller it needs to be converted into digital form. It is done with the help of an ADC convertor which is integrated in the microcontroller.

The above figure exhibits the 5 sensors, microcontroller, Wi-Fi and power supply for each node. The sensors send pulse signal to microcontroller [4, 12] which converts it to machine language. The microcontroller is connected to the Wi-Fi shield so that the data could be sent over the Wi-Fi. The microcontroller is used to interface sensors and Wi-Fi.

6.2. Sending alert to Hospital when Patient Is Critical

Every parameter being monitored has a range such as heart rate, temperature range etc.

Table 2. The Normal Range of Various Body Attributes Being Monitored

Parameter	Heart Rate	Diabetes		Breathing Rate	Body Temperature	Stress
		Without Diabetes	With Diabetes			
Normal Range	60-100 beats per minute	70-99 mg/dl	80-130 mg/dl	12 to 20 breathes per minute	97 F to 99 F	10 μS to 50 μS

These benchmarks are set in the cloud where MQTT protocol resides. If any of the attribute of the person being monitored jumps out of this range, then it would be automatically detected. Due to this an alert containing the GPS coordinates of the patient along with his unique id will get automatically sent to the nearest doctor. This will be achieved by the application [1-3] running in the background. The doctor will be shown an alert stating that a patient needs immediate attention and his location and identity will be in the message transmitted. The unique id will allow the doctor to access the medical history and the readings of the patient of the patient. Due to this the doctor can take prompt action so that the life of the patient could be saved.

6.3. Sending Data to the Application Server

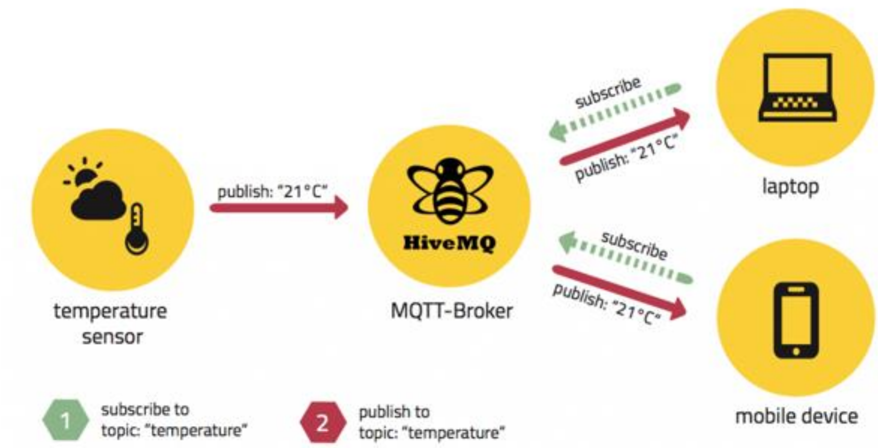

Figure 4. Mechanism of MQTT

The data which are the live readings are sent by the sensors to the application on the smart device using the Wi-Fi connection [6] which is achieved by connecting a Wi-Fi shield [12] with the Arduino. The standard deployed for doing this is MQTT. The MQTT protocol resides in the cloud which acts as a broker for it. MQTT will define methods to indicate the desired action to be performed on the resource. What this resource represents, whether pre-existing data or data that is generated dynamically, depends on the implementation of the server. It waits for a connection to be established with the server. Then the server publishes the values of the sensors to the cloud. The values of the bodily parameters are updated [10] after the following times mentioned in Table 3.

Table 3. Updation Period in Readings of the Body Attributes Being Monitored

Parameter	Heart Rate	Diabetes	Breathing Rate	Body Temperature	Stress
Time	20 secs	15 mins	4 mins	1 hour	40 mins

The resource which is the smart device [15] is being used by the patient subscribing to it. The MQTT [4] will send the data values and will return immediately to the application thread after passing the request to the client [10]. The whole mechanism is shown in figure 6. A sample table is given below to understand how the data for the live readings is stored in the cloud.

Table 4. The Sample Data to show Storage in the Cloud

User Id	Heart Rate (beats per minute)	Diabetes (mg/dl)	Breathing Rate (breathes per minute)	Body Temperature (F)	Stress (μS)
Patient 1	67	78	15	97.4	23
Patient 2	90	110	19	97.8	34
Patient 3	55	88	13	97.5	30
Patient 4	70	72	14	98.9	45
Patient 5	83	92	27	98.4	16

6.4. Application for Doctor and the Patient

The first GUI (on the left hand side) in the above diagram is the application as seen on the patient's smart device. The process begins with the patient registering with their details on the application by selecting the new user option which is displayed at the start of the application. The patient's initial data is stored in the cloud structure [5] which acts as a database. But all the data [16] which is in the form of live readings from their sensors is on another cloud which acts as the broker for MQTT service. So to get this data the application on the smart device is subscribed to cloud MQTT. This is then displayed to the user. The patient can also choose the doctor who he finds best for his treatment. The doctor selected will have access to the patient's details. He can view the patient's current health status and accordingly act. The patient is also given the option of creating a new live appointment. The patients get the option of selecting the date and time. After that, the patients need to wait for the status of their appointment to be approved.

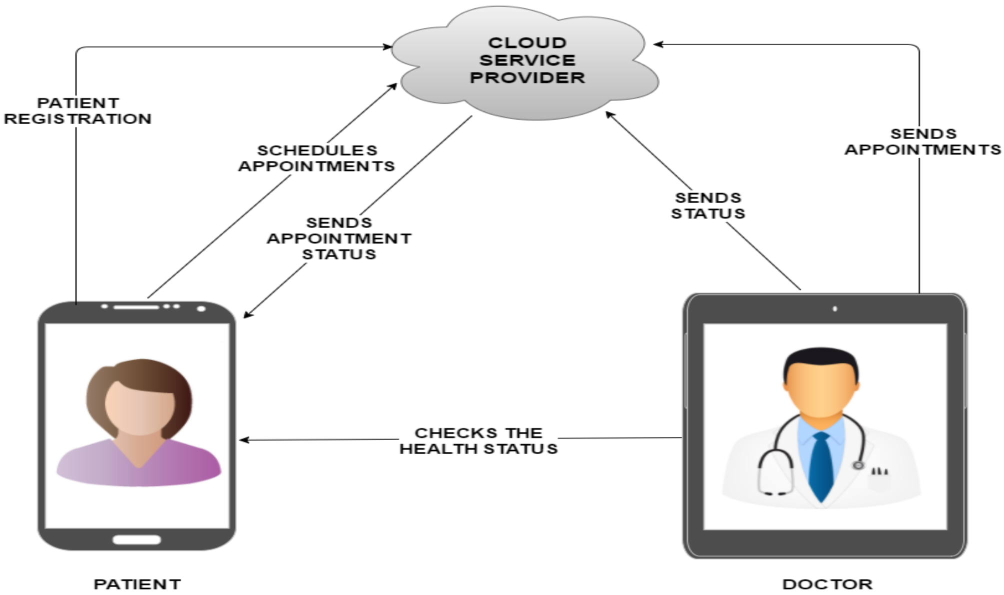

Figure 5. Flow Chart Depicting Application Architecture

The status of the appointment can be checked by the patients by selecting the view appointments option which is displayed immediately after the patient logs into the account. Now once the request for the scheduling the appointment is sent by the patient's side, the doctor's side of the application comes into play. The doctors, similar to the patient have to log in with a username and password. After logging in, the doctors have the option of viewing all the requests for the scheduling of the appointments. They have the option of accepting or rejecting the appointment considering their schedule. Once the appointment is confirmed, the patient gets the status of the confirmation. After this, the live video chat facility will be use for the interaction by the doctor and the patient. This will be done once the appointment is scheduled. The main advantage of the live video chat facility is that the doctor can be consulted from anywhere in the world.

7. Cost Analysis of the Model

It is important that we mention the cost of the entire project. This will make it better understood and justified in terms of feasibility so that the rural areas don't feel the price pinch while going for it. So the cost of the various components in the wearable part is:

Table 5. Price of the Components

Components	Price (in Rupees)
Heart Rate Sensor	500
Body Temperature Sensor	20
Diabetes Sensor	800
Stress Level Sensor	700
Respiratory Sensor	1500
Arduino Uno	500
Wi-Fi Shield	300

The total of these components roughly comes to around Rs.4300. But if the components are bought in bulk then it will cost about Rs.3000 for one whole product. The software used are not priced as the work on them could be done for free. So the overall price is quite reasonable according to rural standards as well and people from all walks of life could use it.

8. Criteria for Evaluation of the Wearable

The characteristics on the basis of which the wearable is evaluated are as follows:

- Range - Every sensor in the wearable operates over a specified range. The design ranges are usually fixed. If by chance they are exceeded then it could result in permanent damage to or destruction of a sensor. The different ranges of the sensor are given in Table-2.

- Sensitivity – : change in output of it per unit change in the parameter being measured. $Senstivity = \frac{TP}{TP + FN}$
- Accuracy–:ratio of the lowest deviation of a value to the ideal value. So the accuracy is given as: $Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$

- Precision – Precision refers to the random variability. It is the closeness of two or more measurements to each other. It is given as : $\Pr ecsison = \frac{TP}{TP + FP}$
- Specificity –defined as its ability to correctly identify those without the disease. It is calculated as follows : $Specificity = \frac{TN}{TN + FP}$

TP (True Positive Rate) = It illustrates the records that are predicted as true as they are actually true.

TN (True Negative Rate) = It illustrates the records that are predicted as false as they are actually false.

FP (False Positive Rate) = It illustrates the records that are predicted as true as they are actually false.

FN (False Negative Rate) = It illustrates the records that are predicted as false as they are actually true.

To show the efficiency of the system we tested the wearable on some patients and then recorded the number of true positives, true negatives, false positives and false negatives. This is shown in the Table 6 given below.

Table 6. Patient Health Indication Instances

	True Positive	True Negative	False Positive	False Negative
Patient 1	4	3	1	1
Patient 2	8	9	1	2
Patient 3	9	11	2	2
Patient 4	5	3	0	1

Patient 5	7	2	3	0
Patient 6	3	3	0	0
Patient 7	10	5	1	1
Total	46	36	8	7

So according to these recordings the sensitivity, accuracy, precision and specificity is as follows –

Sensitivity = 0.86; Accuracy = 0.84; Precision = 0.85; Specificity = 0.82;

These values show that the whole system is rather highly effective and prove to be value for money for its users.

9. Technical Challenges

Problems with the use of this technology could include:

- **Security**: Considerable effort would be required to make transmission secure and accurate. It would have to be made sure that the patient "secure" data is only derived from each patient's dedicated system and is not mixed up with other patient's data. Security is a high priority in most networks. An issue could be created out of confidentiality, authentication, integrity, and freshness of data together with availability and secure management.
- **Interoperability**: The system would have to ensure seamless data transfer across various standards. Further, it would have to be scalable, ensure efficient migration across networks and offer uninterrupted connectivity.
- **Sensor validation**: Sensing devices are subject to inherent communication and hardware constraints including unreliable wired/wireless network links, interference and limited power reserves. This may result in erroneous datasets being transmitted back to the end user. It is of the utmost importance especially within a healthcare domain that all sensor readings are validated. This helps to reduce false alarm generation and to identify possible weaknesses within the hardware and software design.
- **Interference**: The wireless link used for body sensors should reduce the interference and increase the coexistence of sensor node devices with other network devices available in the environment.
- **Constrained deployment**: The system is a wearable, lightweight and non-intrusive. It should not alter or encumber the user's daily activities. The technology should ultimately be transparent to the user. Also it should perform its monitoring tasks without the user realising it.
- **Consistent performance**: The performance of the model should be consistent. Sensor measurements should be accurate and calibrated, even when the wearable is

switched off and switched on again. The wireless links should be robust and work under various user environments.

- **Application Fault:** Sometimes due to load the application may take time to load the readings. Also the application could crash if the operating system is very old and is not suitable for handling the functionality of the app.
- **Cloud Issue:** The performance of the cloud could vary. This would be a problem for data analysis and processing part. Also the data storage may not be so much efficient.

10. Results and Discussion

Hence the whole model could be deployed consisting of the wearable as well as the application communicating with it. Various figures and screenshots are given below to show how the model has been implemented.

10.1. Making and deploying the wearable

The wearable is made using the Arduino and various sensors. These sensors take various types of readings. The figures 6 to 9 display how the wearable is made and how it is attached to the person's body.

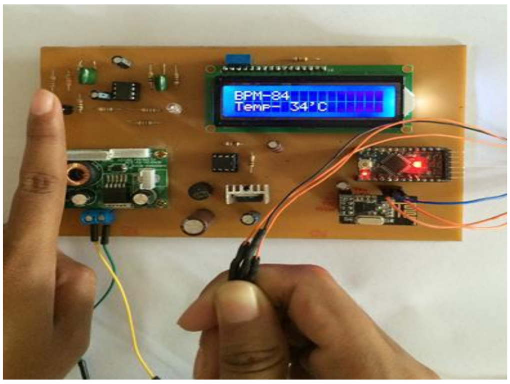

Figure 6. Arduino Live Working

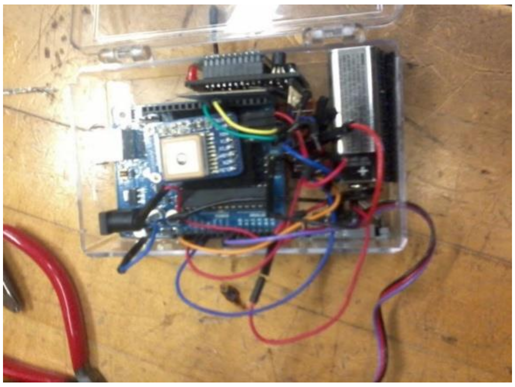

Figure 7. Wearable

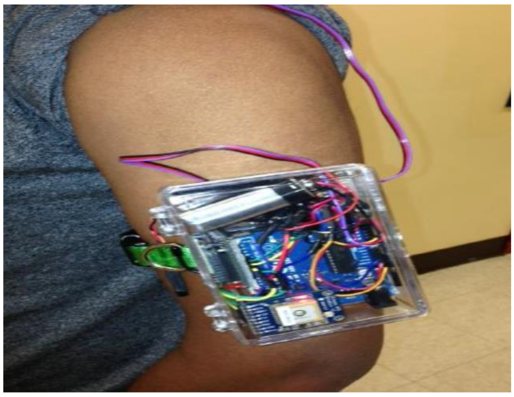

Figure 8. Wearable Attached to the Body

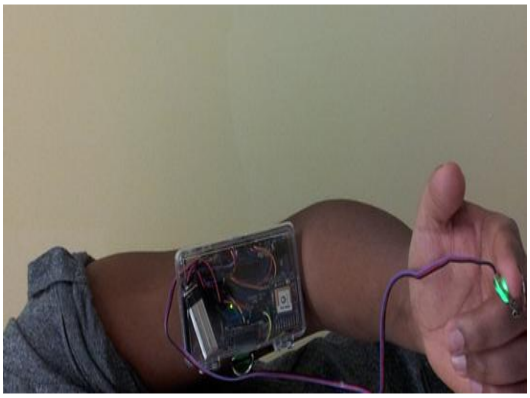

Figure 9. Sensor Node at the Finger

10.2. Emergency Service

In case of emergency when the patient is critical the nearest doctor or healthcare is found out through the help of a background app. Figure 10 demonstrates this concept. The alert is sent to the doctor on his smart device. The alert is displayed as shown in Figure 11.

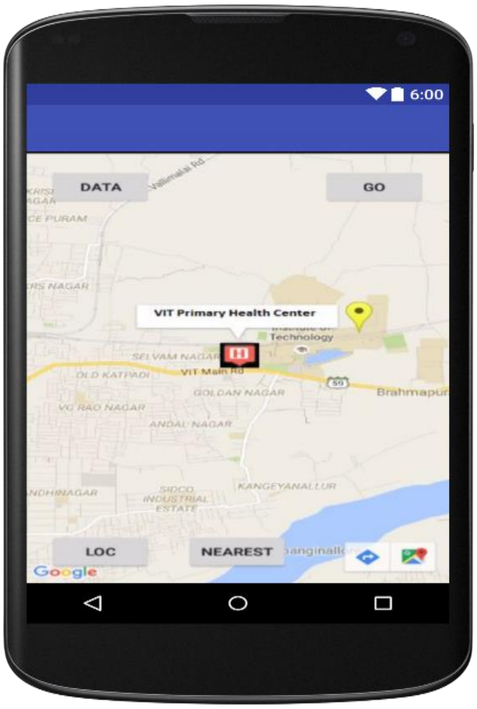

Figure 10. Location of Nearest Health Centre

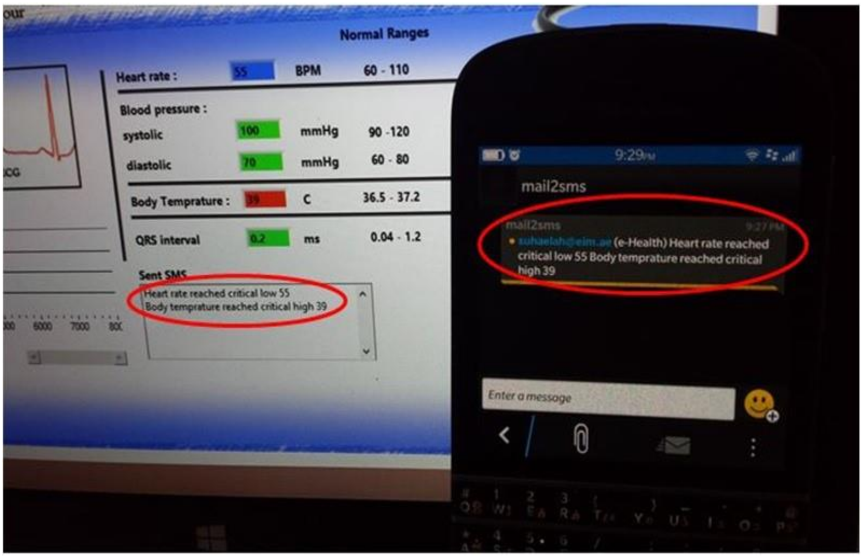

Figure 11. Message Alert on Crossing the Threshold

10.3. Main Application

Patient-Doctor Registration and Appointment Scheduling

The patients have to register in the application. After that they can take a live appointment from the doctor. This is shown in Figures 12-15. Additionally, options like making a phone call, live video chat and checking the patient's health status is given to the doctor on his GUI. This is depicted by Figure 16. The live readings of the patient are displayed in an interactive manner in the application with the help of graph as shown in the Figure 17. This helps the patient to understand and make sense of the readings in a much better manner.

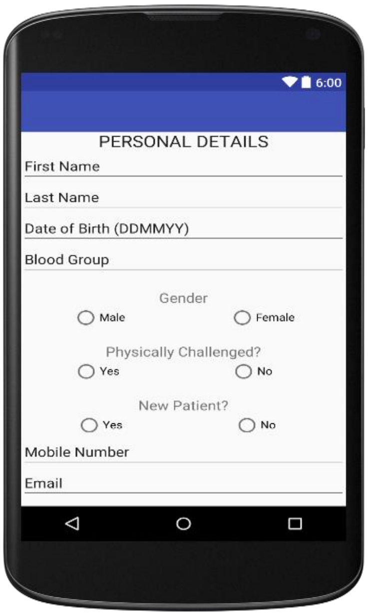

Figure 12. Patient Registration Interface

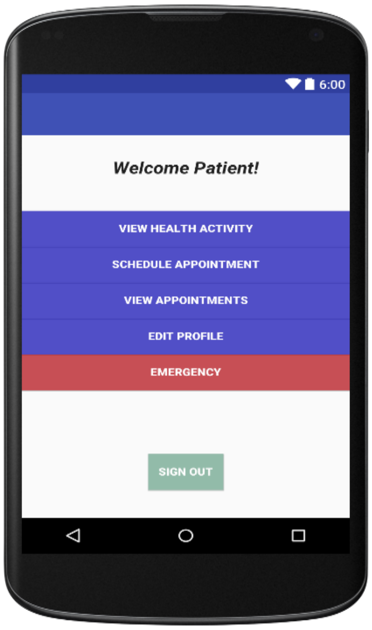

Figure 13. Options for Patient

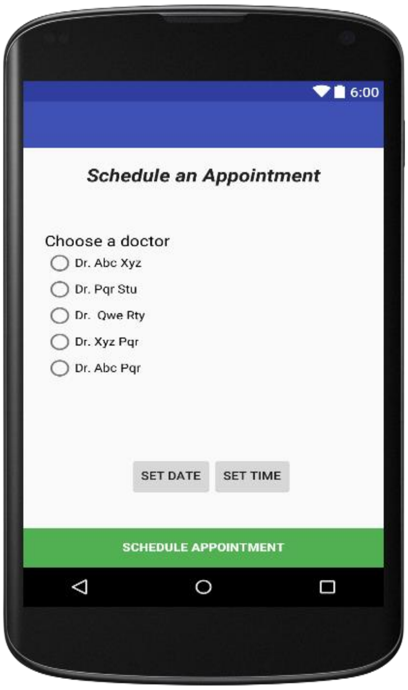

Figure 14. Scheduling an Appointment

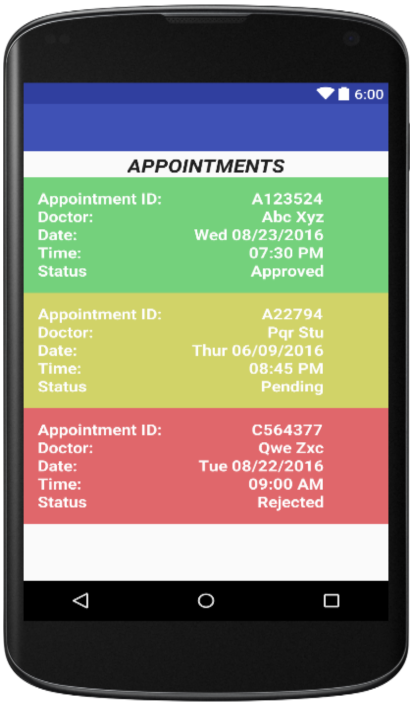

Figure 15. GUI for Patient Appointments

Figure 16. GUI for Video Consultation with Doctor

Figure 17. Live Reading in the Form of Graph

11. Conclusion and Future Works

This robust system is very helpful to the users. It reduced the distance gap between the healthcare centres and the patients at rural villages . Initially used at our university with for about 200 students. It worked well. Anybody could avail it at a low cost which is a major factor for people in rural areas. It would be really facilitative for the elderly or for the people who are suffering from various health issues which limits their movement. Such problems will reduce drastically and it will help provide a better way of living. The traditional method of manually going to the hospital and scheduling the appointments is very tedious, excruciating and enervating. The main advantage of our model is that it connects patient and doctors sitting far away from each other and also saves the patient precious time. Keeping the convenience of the patients in mind, we have created this application in such a way that it can be accessed from anywhere in the world. We aim to expand the application part to the platforms such as IOS and Windows because currently our application can be used only by the users who have an Android phone. Better technologies like ZigBee and Raspberry Pi could be put to use for faster processing and transmission of data.

References

[1] A. Hylton III and S. Sankaranarayanan, "Application of Intelligent Agents in Hospital Appointment Scheduling System", International Journal of Computer Theory and Engineering, vol. 4, no. 4, **(2012)** August, pp. 625-630.

[2] B. M. Lee, "Dynamic Data Binding Protocol between IoT Medical Device and IoT Medical Service for Mobile Healthcare", International Journal of Smart Home, vol. 9, no. 6, **(2015)**, pp. 141-150.

[3] D. Modi, J. Vyas and P. Shah, "Android Based Patient Monitoring System", International Journal for Technological Research in Engineering, vol. 1, Issue 9, **(2014)** May, ISSN(online): 2347- 4718.

[4] H. M. Thwe and H. M. Tun, "Patient Health Monitoring Using Wireless Body Area Network", International Journal of Scientific & Technology Research, vol. 4, Issue 06, **(2015)** June, pp.364-368.

[5] I. Ahmed, S. Janghel and N. Ch. S. N. Iyengar, "Ubiquitous Cloud Based Bookcafé", International Journal of Database Theory and Application, vol.7, no.6, **(2014)**, pp. 63-80.

[6] J. -V. Lee, Y. -D. Chuah and K. T. H. Chieng, "Smart Elderly Home Monitoring System with an Android Phone", International Journal of Smart Home, vol. 7, no. 3, **(2013)** May, pp. 17-32.
[7] K. P. Rao, M. A. Hanash and G. A. AL-Aidaros, "Development of Mobile Phone Medical Application Software for Clinical Diagnosis", International Journal of Innovative Science and Modern Engineering (IJISME), vol. 2, Issue 06, **(2014)** June, pp. 5-8.
[8] K. H. Park and J. G. Park, "Implementation of a Handheld Compute Engine for Personal Health Devices", International Journal of Smart Home, vol. 6, no. 2, **(2012)** April, pp. 59-64.
[9] K. Lee, Y. E. Gelogo and S. Lee, "Mobile gateway System for Ubiquitous system and Internet of Things Application", International Journal of Smart Home, vol. 8, no. 5, **(2014)**, pp. 279-286.
[10] M. Chuah, F. Fu and P. Yang, "Sensor-Based Medical Information System (SBMIS)", International Journal of Smart Home, vol. 2, no. 2, **(2008)** April, pp. 111-126.
[11] P. Sundaram, "Patient Monitoring System Using Android Technology", International Journal of Computer Science and Mobile Computing IJCSMC, vol. 2, Issue 5, **(2013)** May, pp.191 – 201.
[12] R. S. S. Nikilla, "Real Time Health Monitoring System using Arduino", South Asian Journal of Engineering and Technology, vol. 2, no.18, **(2016)**, pp. 52–60.
[13] R. Wei and Z. Yang, "Design and Implementation of Doctor-Patient Interaction System Based on Android", 2012 International Symposium On Information Technology in Medicine And Education, **(2012)** August 3-5, Hokodate, Hokkaido, Japan.
[14] S. B. Choudhari, C. Kusurkar, R. Sonje, P. Mahajan and J. Vaz, "Android Application for Doctor's Appointment", International Journal of Innovative Research in Computer and Communication Engineering, vol. 10, no. 2, **(2011)** November, pp. 2472-2474.
[15] Z. A. Habash1, W. H. W. Ishak and M. H. Omar, "Android Based Application to assist doctor with Alzheimer's patient", 4th International Conference on Computing and Informatics, ICOCI 2013, **(2013)** August 28-30, Sarawak, Malaysia, Universiti Utara Malaysia.
[16] Z. Ali, A. Rahim and S. S. Nawaz, "Design and Implementation of a Low Cost Wireless Sensor Network using Arduino and nRF24L01(+)", Second National Conference on Emerging Trends in Computer Science & Engineering (NCETCSE-2016), **(2016)** April 29, Bengaluru, India.
[17] https://www.abbottdiabetescare.com/
[18] http://www.clouddx.com/vitaliti.html
[19] http://www.embrace.net/
[20] https://www.getqardio.com/en/qardiocore-wearable-ecg-ekg-monitor-iphone/
[21] http://www.healthcareoriginals.com/
[22] http://www.proteus.com/
[23] https://www.quellrelief.com/
[24] https://www.valedotherapy.com/de_en/

Authors

Akarsh Goyal, he is a student, pursuing a Bachelors of Technology in Computer Science and Engineering at VIT University, Vellore, Tamil Nadu, India. He is an avid reader and computer enthusiast. He has published a few scientific papers in area of IoT, data mining, software engineering and marketing. His areas of interest are data mining, machine learning, intelligent systems, IoT, and android based applications. Akarsh has a passion for design which is reflected by his participations in hackathons. He has a particular aptitude to projects with real life application and scope using computer applications.

Ishan Khandelwal, he is a student, pursuing a Bachelors of Technology in Computer Science and Engineering at VIT University, Vellore, Tamil Nadu, India. He is an avid reader and computer enthusiast. Competitive Coding amazes him. He has developed a lot of apps. Some of his apps are also there on play store. He likes to work on real-time systems and internet of things based design models. He has been a part of various hackathons and working on application designs is his forte. His other interests are machine learning and intelligent systems.

Aditya Kumar Mathur, he is a student, pursuing a Bachelors of Technology in Computer Science and Engineering at VIT University, Vellore, Tamil Nadu, India. He is an avid sports enthusiast. He is a team player and has participated in various large group projects. His areas of interest are data mining, IoT, and smart-devices based applications. Aditya is part of many NGOs like Fifth Pillar and Make a Difference. He has a very creative mind and has keen interest in software development management.

Ronnie D. Caytiles, he had his Bachelor of Science in Computer Engineering- Western Institute of Technology, Iloilo City, Philippines, and Master of Science in Computer Science– Central Philippine University, Iloilo City, Philippines. He finished his Ph.D. in Multimedia Engineering, Hannam University, Daejeon, Korea. Currently, he serves as an Assistant Professor at Multimedia Engineering department, Hannam University, Daejeon, Korea. His research interests include Mobile Computing, Multimedia Communication, Information Technology Security, Ubiquitous Computing, Control and Automation.

N. Ch. S. N. Iyengar, he is a Senior Professor, SCOPE at VIT University, Vellore, TN, India. His research interests include cloud Computing, Information Security, Big data, Intelligent Computing, e-Commerce, Health Informatics and Fluid Dynamics (Porous Media). He had 30+ years of teaching and research experience with 200+ publications with good credentials.