THE PERFORMANCE OF IOT BASED HEALTH MONITORING SYSTEM IN VARIOUS ENVIRONMENTS

C. S. GASTON RAVIN DIAS

FACULTY OF ENGINEERING
UNIVERSITY OF MALAYA
KUALA LUMPUR

2019

THE PERFORMANCE OF IOT BASED HEALTH MONITORING SYSTEM IN VARIOUS ENVIRONMENTS

C. S. GASTON RAVIN DIAS

THESIS SUBMITTED IN FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF BIOMEDICAL ENGINEERING

FACULTY OF ENGINEERING
UNIVERSITY OF MALAYA
KUALA LUMPUR

2019

UNIVERSITY OF MALAYA

ORIGINAL LITERARY WORK DECLARATION

Name of Candidate: C. S. Gaston Ravin Dias

Matric No: KQB170006

Name of Degree: Masters in Biomedical Engineering

Title of Thesis ("this Work"): The Performance of IoT Based Health Monitoring System in Various Environments

Field of Study: Applications of Internet of Things (IoT) in Healthcare

I do solemnly and sincerely declare that:

(1) I am the sole author/writer of this Work;
(2) This Work is original;
(3) Any use of any work in which copyright exists was done by way of fair dealing and for permitted purposes and any excerpt or extract from, or reference to or reproduction of any copyright work has been disclosed expressly and sufficiently and the title of the Work and its authorship have been acknowledged in this Work;
(4) I do not have any actual knowledge nor do I ought reasonably to know that the making of this work constitutes an infringement of any copyright work;
(5) I hereby assign all and every rights in the copyright to this Work to the University ofMalaya ("UM"), who henceforth shall be owner of the copyright in this Work and that any reproduction or use in any form or by any means whatsoever is prohibited without the written consent of UM having been first had and obtained;
(6) I am fully aware that if in the course of making this Work I have infringed any copyright whether intentionally or otherwise, I may be subject to legal action or any other action as may be determined by UM.

Candidate's Signature Date:

Subscribed and solemnly declared before,

Witness's Signature Date:

Name:

Designation

THE PERFORMANCE OF IOT BASED HEALTH MONITORING SYSTEM IN VARIOUS ENVIRONMENTS

ABSTRACT

Nowadays, ‘Internet of Things’ (IoT) is playing an important role in all industries and soon it will become a dominant application which will influence our day-to-day life. IoT means the interconnections of all the devices what we use in our day-to-day life via the internet which will provide the facilities of operating, controlling, data collection and analyzing of those devices from anywhere around the world. In these modern days, due to the booming growth in Healthcare Industry, many researches are carried out on these IoT concept and several techniques have been widely applied in order to provide best medical assistance with cutting edge technology to provide less effort and low cost healthcare services. ‘Telemedicine’ and ‘Remote Patient Monitoring’ are some other techniques used in the healthcare industry which comes under the applications of IoT in Healthcare. Basically, all these concepts help to monitor the patients without being with them physically or in other words ‘Remote Monitoring’, which gives enough space for both patient and healthcare providers and many more advantages physically and economically. In this project, performances of IoT based Healthcare Monitoring System in various environments are studied. The related techniques in the literature are reviewed in order to understand the existing techniques and the functions of such systems. A prototype of a standard Remote Patient Monitoring system has been developed and tested in various environments. Results have indicated that the applied Network Type and its Signal Strength have impacts on reducing the effectiveness of such Systems. Ways to improve the system performance are discussed in this research report.

Keywords: - Internet of Things, Healthcare, Remote Patient Monitoring.

PRESTASI SISTEM PEMANTAUAN KESIHATAN BERASASKAN IOT DALAM PELBAGAI PERSEKITARAN

ABSTRAK

Kini, 'Internet of Things' (IoT) memainkan peranan penting dalam semua industri dan tidak lama lagi ia akan menjadi aplikasi dominan yang akan mempengaruhi kehidupan seharian kita. IoT bermaksud interkoneksi semua peranti apa yang kita gunakan dalam kehidupan sehari-hari kita melalui internet yang akan menyediakan kemudahan pengendalian, pengawalan, pengumpulan data dan analisis peranti-peranti tersebut dari mana saja di seluruh dunia. Dalam zaman moden ini, disebabkan oleh pertumbuhan pesat dalam Industri Penjagaan Kesihatan, banyak penyelidikan dijalankan pada konsep IoT dan beberapa teknik telah digunakan secara meluas untuk memberikan bantuan perubatan yang terbaik dengan teknologi canggih untuk menyediakan perkhidmatan dengan usaha yang minimum dan perkhidmatan penjagaan kos kesihatan yang rendah. 'Telemedicine' dan 'Pemantauan Pesakit Jauh' adalah beberapa teknik lain yang digunakan dalam industri penjagaan kesihatan yang terdapat di bawah aplikasi IoT dalam Penjagaan Kesihatan. Pada asasnya, semua konsep ini dapat membantu memantau pesakit tanpa mereka perlu ada secara fizikal atau dengan kata lain 'Remote Monitoring', yang memberikan ruang yang cukup untuk pesakit dan penyedia penjagaan kesihatan serta banyak lagi kelebihan dari segi fizikal dan ekonomi. Dalam projek ini, persembahan IoTSistem Pemantauan Penjagaan Kesihatan dalam pelbagai persekitaran dipelajari. Teknik yang berkaitan dalam kesusasteraan dikaji semula untuk memahami teknik yang sedia ada dan fungsi sistem sedemikian. Satu prototaip sistem Pemantauan Pesakit Secara Jauh yang standard telah dibangunkan dan diuji dalam pelbagai persekitaran. Keputusan menunjukkan bahawa Jenis Jaringan yang digunakan dan Kekuatan Isyaratnya mempunyai kesan ke atas mengurangkan keberkesanan Sistem tersebut. Teknik untuk meningkatkan prestasi sistem dibincangkan dalam laporan penyelidikan ini.

Kata kunci : - Internet Perkara, Penjagaan Kesihatan, Pemantauan Pesakit Jauh.

ACKNOWLEDGEMENTS

First of all, I would like to thank the Almighty for the time and grace given to me for these two years of post graduate studies. Secondly, my heartfelt gratitude to my father who is supporting me from all sides for my studies. Thirdly, I would like to thank, Dr. Mohd Yazed Bin Ahmad who was my supervisor for this Research Project. He was an amazing person to give technical ideas and guided me thoroughly, throughout this entire research project report preparation. He was so kind and flexible to me and supported me to prepare this report in between my busy schedules. Definitely I should mention my dear brother, Mr. C. S. Winston Navin Dias who is an Electronics and Telecommunication Engineering student who helped me a lot to prepare coding and gave several technical ideas for this project. Last but not least I would like to thank all my friends who supported me during this entire project work.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS AND ABBREVIATIONS

ADC	:	Analog to Digital Converter
IDE	:	Integrated Development Environment
IoT	:	Internet of Things
BSN	:	Body Sensor Network
ECG	:	Electrocardiogram
EMG	:	Electromyogram
PCB	:	Printed Circuit Board

LIST OF APPENDICES

CHAPTER 1: INTRODUCTION

IoT is an advanced application of internet connectivity in most physical devices which can be synchronized and controlled remotely. Nowadays these concepts are widely used in each and every field and make the human life more sophisticated ever than before, by connecting all smart devices and resources we use in our day to day life through the Internet. Applications of this IoT in Healthcare services are a new concept in Healthcare sector, thus it is more focused and many researches are conducted in these area. Due to increasing population which triggers economical crisis, increasing amount of chronic disease patients, lack of knowledge to maintain a balance and healthy life, inability to get immediate medical access, living below the poverty line, lack of healthcare providers in remote areas are some major factors which urge to focus on Telemedicine and Remote Patient Monitoring where this IoT concept is successfully applied.

1.1. Overview.

Patients with chronic illnesses such as diabetes, heart disease, arthritis, kidney disease, HIV/AIDS, lupus, multiple sclerosis, etc… are need to be treated for a long term and monitored frequently for any sudden risks. Poverty, lack of transportation services in remote areas, physical impairments to travel and go to hospitals, long queues and waiting lists in hospitals and lack of healthcare providers are some facts which make the patients of rural areas of developing countries to face certain risk conditions such as sudden heart attacks, strokes, memory loss and getting coma stage before getting the proper treatments. Even the medical aids are provided after facing such worst conditions; there are possibilities to lose lives due to late medical treatments after those worst conditions. The major reasons for people of developed countries also faces such risks even though they have all medical facilities; are because of lack of time to take care of their own health due to the busy work and stressful life style they live in. Therefore Telemedicine and Remote Patient Monitoring with the application of IoT concept are widely welcomed by the patients with great hope and in near future this will be a very important practice in all over the healthcare industry.

Due to some major advantages of these concepts such as monitoring and consulting a patient remotely from anywhere around the world, healthcare providers can work from their convenient places like from home and patients can save a lot of time and money which spent unwontedly during the waiting time at the hospital and paying for hospital utilities. Most of the patients in near future will prefer to get these kinds of monitoring services from their healthcare providers.

1.2. Problem Statement.

There are many researchers have been carried out to develop such 'Telemedicine' and 'Remote Patient Monitoring' concepts. Since the patients are monitored remotely at different locations (i.e. hospitals, home, office) by the Body Sensor Network and the medical data travel through a wide range of monitoring and processing devices, there are many possible factors which have influence while handling the medical data which may cause disadvantage effects (high delay rate, high noise, low signal strength) in Real-Time Monitoring and this will make the situation bad if the patient is facing any high risk conditions where real time monitoring is vital.

1.3. Aim.

Study the operation of IoT based wireless health monitoring system in various environments and identifies the factors that influence the system performance and investigate the ways to improve the performance.

1.4. Objectives.

I. To establish a standard IoT based wireless health monitoring system.

II. To operate the system under different environments and perform data collection such as time delay, network signal strength, environmental conditions and interferences.

III. To analyze the collected data and identify the factors those influence the system performance and suggest ways to minimize such factors and to improve system performance.

1.5. Scope of the Research.

I. Entire research uses a standard IoT Based Health Monitoring System.

II. Inability to test in multiple environments (i.e.: while bathing, Market place, LRT Station, Worship places)

III. Only two parameters of vital signs are considered due to limited resources (Heart Rate and Body Temperature).

IV. This Monitoring System uses less expensive or free facilities (low cost sensors, modules, free cloud service, free university Wi-Fi service and personal mobile hotspot service) which limit the outcomes of the results.

CHAPTER 2: LITERATURE REVIEW

There are a lot of researches have carried out on the concept of 'Applications of IoT in Healthcare' during the last couple of decades. A certain number of such literatures are reviewed here in order to find the research gap which will support to develop a proper methodology for this Research Project.

2.1. Implementation of IoT Concepts in Healthcare.

The latest technologies in recent years allow us to synchronize multiple devices which we use in our day to day life for communication and gathering knowledge. The IoT in healthcare is a new concept and application where such devices along with additional sensors to collect real time data of our vital signs (Ullah et al., 2016). In IoT based Healthcare Monitoring, several devices gather medical data and send them to Cloud in real time where a large amount of data can be collected, analyzed and stored in several forms and can activate conditional alarms (Kodali et al., 2010).

Enable user to improve from health related risks and reduce the costs for the healthcare services by collecting, storing, analyzing and sharing large amount of medical data in real time with high efficiency. And also to reduce the tensions felt by patients while visiting a doctor every time to check their vital signs (Gupta et al., 2016). Furthermore with the help of this system the time of both the patient and doctor can be saved a lot and the doctor can provide his service especially for emergencies via remotely at his full. Using the smart phones in order to add more monitoring sensors which can provide more benefits while getting the vital signs of a patient (Kodali et al., 2016).

Cloud based servers are used generally in order to save, analyze and provide medical data to the accessing person (healthcare providers) from anywhere around the world through their smart devices such as smart phones, laptops, tablets, etc… via the internet connections for remote monitoring and consultations (Gupta et al., 2016; Ullah et al., 2016; Kodali et al., 2016; Wu et al., 2017). According to Gope et al. (2015), Body Sensor Network technology is one of the major applications of IoT in healthcare. In the applications of IoT, mainly radio frequency identifications, wireless sensor networks and smart phone technologies are well trending (Catarinucci et al., 2015).

The ever-increasing advancement in communication technologies of modern smart objects brings with it a new era of application development for Internet of Things (IoT)-based networks. In particular, owing to the contactless-ness nature and efficiency of the data retrieval of mobile smart objects, such as wearable equipment or tailored bio-sensors, several innovative types of healthcare systems with body sensor networks (BSN) is needed. Therefore a secure IoT-based healthcare system, which operates through the BSN architecture has been proposed here (Yeh., 2016).

(Doukas et al., 2012) propose Pervasive healthcare applications utilizing body sensor networks generate a vast amount of data that need to be managed and stored for processing and future usage. Cloud computing among with the Internet of Things (IoT) concept is a new trend for efficient managing and processing of sensor data online. Their work presents a platform, based on Cloud Computing for management of mobile and wearable healthcare sensors, demonstrating this way the IoT paradigm applied on pervasive healthcare.

Nowadays, ageing related diseases represent one of the most relevant challenges for developed countries. The use of healthcare remote technology may allow reducing most of the management of the chronic diseases meanwhile it may also contribute to the improvement of elderly people’s quality of life. Unfortunately, despite the advent of Internet of things and the even decreasing price of sensors, current proposals are not extensible during runtime meaning that they need to be maintained offline by engineers. Therefore, Jimenez et al., (2015), discuss how to build an ad-hoc extensible (during runtime) healthcare monitoring system by using low cost wireless sensors and already existent Internet of things technology as communication platform. Moreover, they present a prototype of a basic healthcare remote monitoring system, which alerts, in real time, patients' relatives or medical doctors that an elderly people is experienced a problem that could need medical attention or hospitalization.

The intelligent use of resources enabled by Internet of Things has raised the expectations of the technical as well as the consumer community. However there are many challenges in designing an IoT healthcare system, like security, authentication and exchanging data. The IoT healthcare system, is transforming everyday physical objects, medical devices that surround us into an embedded smart healthcare system. Public healthcare has been paid an increasing attention given the human population and medical expenses exponential growth. It is well known that an effective monitoring healthcare system can detect abnormalities of health conditions in time and make diagnoses according to sensing (WBSN) data. F. Nasri et al., (2017), propose a general architecture of a smart mobile IoT healthcare system for monitoring patients risk using a smart phone and 5G. The design of multi-protocol unit for universal connectivity. Web and mobile applications developed to meet the needs of patients, doctors, and laboratories analysis and hospitals services. The system advises and alerts in real time the doctors/medical assistants about the changing of vital parameters of the patients, such as body temperature, pulse and Oxygen in Blood etc… and also about important changes on environmental parameters, in order to take preventive measures, save lives in critical care and emergency situations.

2.2. Types of Available Systems.

The work of Kodali et al. (2010), explains the implementation of an IoT based In-hospital healthcare system using ZigBee mesh protocol. This system can monitor periodically the vital signs of the in-hospital patients. (Gupta et al., 2016), deals with IoT based monitoring system for emergency medical services. Heart Rate and Body Temperature are the two medical data collected from the patient and doctors can monitor them remotely through this system. Various other sensors like Blood Pressure Sensor, ECG Sensor, etc can be added later to this Smart Healthcare Kit, according to the patient's requirements.

Ullah et al. (2016), proposes an e-Health (healthcare practice via electronic devices) model for patients. It is named as 'k-Health'. This k-Health model is built by four different layers which are Sensor Layer, Network Layer, Internet Layer and Service Layer which all can cooperate with each other to provide a platform from accessing patients' health data till providing healthcare support. Kodali et al. (2016), proposes the system 'Real Time Healthcare Monitoring System Using Android Mobile'. Wu et al. (2017), proposes wearable sensor nodes (each wearable sensor is considered as a single sensor node) with solar energy harvesting and Bluetooth Low Energy (BLE) transmission that enables the implementation of an autonomous Wireless Body Area Network (WBAN).

Body Sensor Network where patients can be monitored by tiny powered and light weighted sensor nodes (Gope et al., 2015). Rohokale et al. (2011), proposes a cooperative IoT approach for a better health and monitoring and control for rural people's health parameters like blood pressure, glucose level, cholesterol level, etc… IoT aware smart architecture for automatic monitoring and tracking of patients, personal and biomedical devices with in the healthcare service delivering premises. The proposed

Smart Hospital System which depends on multiple technologies such as Radiofrequency Identifications, wearable sensor networks and smart phones which all can interoperate with one another via a constrain application protocol over low power wireless personal area networks / representational state transfer network infrastructure (Catarinucci et al., 2015).

Doukas et al., (2012) presents a platform of cloud based computing which is for management of mobile and wearable body sensors. Jimenez et al. (2015), discusses how to build an ad-hoc extensible healthcare monitoring system by using less expensive wireless sensors along with IoT technology as its communication platform. Apart from this they discuss a basic heakth monitoring health system which can alert in real time to patient's relatives or healthcare providers whenever an elderly person (who is attached with this system) is facing a trouble which may need immediate medical assistance. Gia et al. (2014), presents an IoT system which is cost effective and having easy way to analyze and monitor, either remotely or at the spot of the real time ECG and EMG.

Internet of Things (IoT) has paved a path towards the digitization of everyday things connecting each other through internet. Due to the huge advent of IoT in recent years, researches have started to accomplish the long cherished will of human being to make life simpler and better in many ways. Health being the most valuable wealth of human should be given most priority. Though health related research implying IoT has been neglected due to heterogeneity and interoperability issues. Ray (2014), presents 'H3IoT' a novel architectural framework for Home Health Hub Internet of Things for monitoring health of elderly people at home. The framework is promising in terms of its design and future envision of usage of real life implementation for H3IoT.

In the study of C.C. Lin et al., (2018), an intelligent health monitoring system based on smart clothing is proposed. The system consisted of smart clothing and sensing component, care institution control platform, and mobile device. The smart clothing is a wearable device for electrocardiography signal collection and heart rate monitoring. The system integrated our proposed fast empirical mode decomposition algorithm for electrocardiography denoising and hidden Markov model–based algorithm for fall detection. Eight kinds of services were provided by the system, including surveillance of signs of life, tracking of physiological functions, monitoring of the activity field, anti-lost, fall detection, emergency call for help, device wearing detection, and device low battery warning. The performance of fast empirical mode decomposition and hidden Markov model were evaluated by experiment I (fast empirical mode decomposition evaluation) and experiment II (fall detection), respectively. The accuracy and sensitivity of R-peak detection using fast empirical mode decomposition were 96.46% and 98.75%, respectively. The accuracy, sensitivity, and specificity of fall detection using hidden Markov model were 97.92%, 90.00%, and 99.50%, respectively. The system was evaluated in an elderly long-term care institution in Taiwan. The results of the satisfaction survey showed that both the caregivers and the elders are willing to use the proposed intelligent health monitoring system. The proposed system may be used for long-term health monitoring.

The development on newer healthcare services for the elderly citizens has become an immediate necessity today. There have been distinctive health challenges focused in the society through technical innovations. Most of the elderly people today experiences loneliness and psychological depressions, either as a result of living alone/ abandonment or due to reduced connection with their children and relatives. To enhance the quality of services in the elderly healthcare system; H. Basanta et al., (2016), developed a ubiquitous intuitive IoT-based Help to You (H2U) healthcare system to integrate various technologies of wearable devices, biosensors and wireless sensor networks in order to provide an intensive service management platform. This method would support the real time activity and monitor the healthcare system for the elderly citizens. In this method the information collected by various wearable devices in real time are stored in the central database which thereby connects people, doctors, and practitioner at the time of an emergency for the right information. This way the system could increase accessibility, efficiency, and also lower the health expenses to improve the comfort and safety as well as management of daily routines of an elderly life.

Telemedicine is producing a great impact in the monitoring of patients located in remote nonclinical environments such as homes, military bases, ships, and the like. A number of applications, ranging from data collection, to chronic patient surveillance, and even to the control of therapeutic procedures, are being implemented in many parts of the world. As part of this growing trend, K. M. Alajel et al., (2005), present a real-time remote patient monitoring service through World Wide Web (WWW), which allows physicians to monitor their patient in remote sites using popular Web browser. A prototype system is composed of data acquisition and preprocessing module connected to the computer as the remote site via its RS-232 port, two personal computers equipped with network and analog to digital cards, and software modules to handle communication protocols between data acquisition module and personal computer. The purpose of the system is the provision of extended monitoring for patients under drug therapy after infarction, data collection in some particular cases, remote consultation, and low-cost ECG monitoring for the elderly.

2.3. Technology Behind the Available Systems.

In-hospital healthcare system using ZigBee mesh protocol has LM-35 Temperature sensor for getting body temperature signal from the human body. For data transmission from the sensor, a mesh network is created by using XBee Series 2 XB24-B modules which support ZigBee mesh protocol. Indoor range of data transmission of these XBee

Series 2 modules is 40 meters and outdoor range of Radio Frequency Line of Sight Range is 120 meters, which means that this is the maximum distance where transmitters and emitters are placed in view of each other without any obstacles in between. As a central processor, Intel Galileo Generation 2 board is used which receives data from the sensor through the XBee modules and sends them to a cloud server where the medical data will be gathered, analyzed, stored and the processed data can be retrieved remotely by any connected devices used by healthcare providers to this cloud server via internet (Kodali et al., 2010).

This research intends to enhance the quality of care with regular monitoring and reduce the cost of care by digital monitoring (nurses are no needed for monitoring) and actively engage in data collection and analysis. Since the patient monitoring is took place remotely by this IoT application; using only one sensor is not enough to monitor the entire patient's health and health care providers are unable to come up for any decisions only based on body temperature. Since this project is based on In-Hospital Patient Monitoring, ZigBee mesh protocol is acceptable for communications within the hospital premises due to its small, low-power digital radius. But if the Patient Monitoring Range exceeds, another suitable communication protocol may be required accordingly. Apart from this applying this IoT concept eliminates the need for healthcare professionals come and check the patient at regular intervals by providing remote monitoring system via the sensors, network protocols and cloud based analysis and store medical data and display them at anywhere to the healthcare provides via internet enabled devices like personal computers, tablets, smart phones, etc… who can prescribe proper medical treatments.

The system proposed by Gupta et al. (2016), is having 2nd Generation Intel Galileo Board acts as the central processing unit. It collects all the data from the sensors connected to the patient and uploads them to the XAMPP web server via Ethernet. Not only the doctors but also the patient can keep track of the medical data as a web client of this server which can provide live feed backs of the patients' medical conditions as well as maintaining 24 hour records of multiple patients. The Admin page of the web portal allows users to enter personal details which are patient name, age, blood group and other important details in order to keep the record systematically and login to the web portal can be done by both doctor and patient. Apart from these features, doctors can manually add patient's records and descriptions for future references and records can be retrieved even in plotted graph forms for easy observations.

This work intends to improve early health related risk detection, Reduce healthcare costs, Reduce patients' time spent in clinics waiting for doctors in their regular checkups and Giving possibilities to doctors to assist emergency cases as early as possible by this healthcare kit which acts in collecting, storing, analyzing and transmitting large amount of medical data efficiently in real time via XAMPP web server. Live medical condition feed backs and 24 hours of multiple patient history storage, doctors and patients both can access the medical data, sophisticated web page to keep the patient records properly and manual access for doctors to make additional descriptions are the advantages of this system. Moreover obtaining the medical values in the form of graphs will highly helpful to analyze easily and quick decision making for the healthcare providers. Wired connection (Ethernet) to the web server is a disadvantage of this application since wired connections reduce the mobility range of the patients and it is not sophisticated to carry in day to day activities.

According to Ullah et al. (2016), the heart of this model is the Sensor Layer (Wearable Sensor Network-WSN) which consists of various types of sensors such as ECG Sensors, Pulse Oximeter, Blood Oxygen Sensor, etc… along with a smart phone which also has some more different types of sensors, e.g., Accelerometer, Gyroscope, Proximity, Barometer, Temperature, etc… These wearable body sensors gather data and send it to the next level Network Layer provided by the patients' smart phone using various Wireless Personal Area Network (WPAN) protocols (set of rules and procedures for communication between electronic devices). Then the Network Layer transmits all the obtained medical data (body sensors + smart phone sensors) to the third stage, Internet Layer using different TCP/IP protocols or telecommunication technologies and standards like 3G, 4G, ADSL, DSLAM and Router which connects it to the internet via Ethernet or Wirelessly. Internet Layer provides a space for data storage and management by centralized cloud storage. The last and the fourth layer, Service Layer provides direct access of medical data at anytime, anywhere to the healthcare professionals as well as to the patients for delivering health supports remotely and self monitoring respectively.

Structuring the four different layers of this entire system is very helpful for understanding the whole system and its working mechanism. Making use of the smart phone for providing the Network Layer is a novel framework and it is advantageous since no additional network providing modules are needed and using the smart phone's sensors along with the wearable body sensors will reduce the number of buying additional sensors. Due to smart phones play important roles in this concept, it is not possible to apply this monitoring method for those who do not have a smart phone. Likewise remote monitoring will be widely accepted if it uses only wireless methods since it provides high conformability in day to day life. Therefore Ethernet technologies are not scope full for these types of applications.

Kodali et al. (2016), system is based on LPC2148 Microcontroller, monitors the vital parameters such as ECG, Temperature and Heart Rate and transmits the processed data wirelessly through ZigBee technology along with the patient ID so the system can differentiate the different data coming from different patients. The transmitted medical data is displayed in a PC based application called 'The Central Nurse Station' where PC with the receiver ZigBee acts as a hub. At the same time these data is uploaded into cloud based database continuously which can be fetched by authorized healthcare professionals from anywhere at any time. From that database, an Android application fetches all the updated data and displays to the doctors which enables them to receive the complete profile with the current status of the patient in real time. If any parameter of a patient goes beyond a pre-assigned threshold value, an automated notification will pop up in doctor's Android mobile application as a primary alert. If the doctor did not notice that notification, then the Android app can initiate a phone call to make sure that the doctor gets the alert of the patient's condition. Additionally, this system has the ability to notify the nurse station for emergency assistance.

Since ZigBee technology is used for data exchange, its data transmission and receiving is limited within a certain range which is higher than a Bluetooth module and lesser than a Wi-Fi module. Therefore this application of this system is limited by the transmitter, receiver distance. But it is acceptable since the primary data collection is taken place at the Central Nurse Station which must be located within the range of data transmission. While nurses monitor the patient details, those data is uploaded to the cloud based database which can be accessed by authorized people (doctors, parents, friends, etc.) at anytime any where is adding more advantage for this system due to double observation occurs to a single patient. Apart from this, the doctors get updated patient details in real time in their android phones which make them to monitor the patient without being physically present there near to the bed and moreover a doctor can monitor a patient in a different country or different region via this setup which adds more advantages to this framework. Pre-assigning medical readings, notifying the nurse station, alerting the doctors with primary and secondary alerts during emergency situations are really helpful for delivering a better care. Updating this system to other phone operating systems like Apple (iOS), Windows, Black Berry, etc. will be more helpful since doctors use various mobile phones.

As stated in Wu et al. (2017), the main components of these wearable sensor nodes include a Micro Controller Unit and 3 sensors for subject's data collection. ATmega328P is used as the MCU in this system which is the core of this wearable sensor network and is used to collect and process the sensor data, as well as perform power management to reduce the overall power consumption. Multiple sensor nodes are deployed on different positions of the body to measure the subject's body temperature distribution, heartbeat, and fall detection. ADXL362 Accelerometer is used to detect the falls. MAX30205 is used as a body temperature sensor. A commercial pulse sensor is used for heart beat monitoring. Whenever a fall is detected or other emergency conditions occur, an emergency notification will be sent to the smart phone of the medical assistant through the BLE module (HM-10). This BLE module can communicate with most of the iOS (operating system for Apple products) and Android smart phones with Bluetooth 4.0 on the market. For the sensor data visualization, a web-based smart phone application using the Evothings platform is developed. Every smart phone with the 'Evothings Viewer' application installed can access the sensor data through the designed application. This will help the medical assistant or family member monitor the subject's health conditions continuously. To extend the lifetime of the wearable sensor nodes, a flexible solar energy harvester with an output-based Maximum Power Point Tracking (MPPT) technique is used to power the sensor nodes. The harvested energy from the flexible solar panel is stored in a super capacitor.

Harvesting solar energy to power up the sensor nodes is a novel idea in this concept. Flexible small size solar panels for each wearable sensor, MPPT technique is used to maximize power extraction under all conditions to extend the monitoring time by the sensors are benefits here. All of those (doctors, family members, etc.) who have 'Evothing Viewer' in their smart phones can be accessed to the patients' details continuously. This helps to get emergency notifications whenever the patients go through critical stages. Since Bluetooth module is applied here for data transmission, this project can only be applied within a very short range according to the HM-10 module's data transmission range; though solar panels are used; they need to be recharged at outdoor from 23 – 130 minutes as stated by this project under different weather conditions. During this time the effectiveness of the sensor nodes is very less which are some disadvantages of this system.

Generally, BSN systems consist of in-body and on-body sensor networks. In-body sensor networks allow the implanted devices to communicate with the base station and the on-body sensor network is responsible for communication among wearable devices to the monitoring station. Each sensor node is having biosensors such as Electrocardiogram, Electromyogram, Blood pressure, etc… The sensors gather data and send them to Local Processing Unit, which can be smart device like smart phone or tablet. This local Processing Unit acts as a router in between the BSN nodes and the central server named as BSN-Care server via the wireless communication methods such as 3G, 4G, etc… apart from this system is capable to alert the patient or person if any abnormalities detected (like beep tones in mobile phone) in the worn sensors. When the BSN-Care server gets data from the wearable sensors from the local processing unit; then it transmits the data to a database for analyzing. At the same time, if any abnormalities detect at the sensor readings, system will contact the nearby healthcare providers or a family member. Rather than a patient wearing this system, a healthy individual also can wear this to monitor his / her daily vital checks and to keep up healthy habits (Gope et al., 2011).

Rohokale et al. (2011), proposes a way where, a registered person at the rural healthcare center will wear a RFID sensor. The person who wore the sensors will form an Opportunistic Large Array which is a type of a network node structure and the urgent information will be routed via the sink node to the gateway computer and via internet it will be accepted by the rural healthcare doctor. With no internet, this network facility can be used to transmit the information faster. Blood pressure, glucose level, etc… are the health parameters which are considered here. The remote health center monitoring person can update medical data frequently about the provided medications and generate an updated report.

As discussed in Catarinucci et al. (2015), smart hospital system can collect environmental conditions as well as the physiological parameters of the patients in real time through ultra low power hybrid sensing network along with 6LoWPAN nodes integrated with UHF RFID functions. Sensed data via those sensors will be delivered to a central control station where an advanced monitoring application makes them access easily by both local and remote users through a REST web server.

According to Jimenez et al. (2015), the applications can be assessed by different healthcare providers such as doctors, nurses, other medical staffs, etc… the CC2451 Bluetooth sensor gathers ambient data such as temperature, pressure, accelerometer, gyroscope, manometer and humidity with a long lasting battery. Smart phone is used as a gateway for data collection from the worn sensors of the patient at home. The advantage of XMPP server which use the Clayster technology is, allowing installations of new sensors at any time which can be in use for the healthcare providers immediately and also these sensors can be paired with other nearby patients as well.

As provided in Gia et al. (2014), the efficient customization of 6LoWPAN network for medical data gives effective energy consumption and reliable data transmission in various scenarios which needs various healthcare applications.

2.4. Summary of the Literature.

No	Authors' Names.	Proposed System	Components Used
1	Kodali et al., (2010)	IoT based In-hospital healthcare system using ZigBee mesh protocol	XBee Series 2 XB24-B modules, LM-35 Temperature sensor, Intel Galileo Generation 2 board
2	Gupta et al., (2016)	IoT based monitoring system for emergency medical services	2nd Generation Intel Galileo Board, Heart Rate Sensor, Body Temperature Sensor, XAMPP web server
3	Ullah et al., (2016)	k-Health	Smart phone, ECG Sensors, Pulse Oximeter, Blood Oxygen Sensor, etc…, Centralized cloud storage
4	Kodali et al., (2016)	Real Time Healthcare Monitoring System Using Android Mobile	LPC2148 Microcontroller, ECG sensor, Temperature sensor, Heart Rate sensor, ZigBee module, LCD Screen, Cloud based database, Android application
5	Wu et al., (2017)	Autonomous Wireless Body Area Network	Micro Controller Unit - ATmega328P, ADXL362 Accelerometer, MAX30205 temperature sensor, Pulse sensor, HM-10 BLE Module, Smart phone, Evothings platform
6	Gope et al., (2015)	BSN-Care	ECG sensor, EMG sensor, Blood pressure sensor, Mobil networks, BSN-Care server
7	Rohokale et al., (2011)	Cooperative IoT for Rural Healthcare	RFID sensor, glucose sensor, blood pressure sensor, Opportunistic Large

		Monitoring and Control	Array, computer, mobile phone
8	Catarinucci et al., (2015)	Smart Hospital System	RFID, WSN, mobile phone, 6LoWPAN nodes, CoAP, REST web service
9	Gia et al., (2014)	IoT Based Ubiquitous Healthcare System	ECG sensor, EMG Sensor, IPv6, 6LoWPAN, TI CC2538 module with TI ADS1292
10	Doukas et al., (2012)	Cloud Computing towards Pervasive Healthcare	Wearable sensors, mobile phone, Cloud server,
11	Jimenez et al., (2015)	IoT-aware healthcare monitoring system	Bluetooth sensor CC2451 sensor tag, heart rate monitor smart sensor, Smartphone, XMPP server
12	Gia et al., (2015)	Fault Tolerant and Scalable IoT-based Architecture for Health Monitoring	IPv6, 6LoWPAN, TI CC2538 module with TI ADS1292, ECG sensor, EMG Sensor
13	Yeh, K. H. (2016)	A Secure IoT-Based Healthcare System With Body Sensor Networks	Raspberry PI 2, Wearable body sensors such as temperature sensor, accelerometer, heartbeat, Local Processing Unit, Access point, BSN Server.
14	Doukas et al., (2012)	IoT and Cloud Computing towards Pervasive Healthcare	Wearable sensors along with smart phone sensors, Arduino micro controller, Bluetooth module, cloud infrastructure, mobile and web interfaces.
15	Jimenez et al., (2015)	IoT-aware healthcare monitoring system	Smart sensors, smart phones as gateway, Clayster IoT platform
16	P. P. Ray (2014)	Home Health Hub Internet of Things (H^3IoT)	Physiological Sensing Layer (PSL), Local Communication Layer (LCL), Information Processing Layer (IPL), Internet Application Layer (IAL), User Application Layer (UAL)
17	C.C. Lin et al., (2018)	Intelligent health monitoring system based	Smart clothing with sensors, Bluetooth modules, data transmission and

		on smart clothing	receiving modules, cloud platform, mobile device, display at medical unit
18	H. Basanta et al., (2016)	IoT-based Help to You (H2U) healthcare system	Wearable sensors, mobile phone, healthcare cloud database, IoT platform
19	K. M. Alajel et al., (2005)	Remote Electrocardiogram Monitoring Based On The Internet	Data acquisition module, patient's PC as client, doctor's pc as server, cloud database
20	F. Nasri et al., (2017)	Smart Mobile Healthcare System based on WBSN and 5G	Smart wireless body sensors networks, Wireless network communication, Smart phone, web data base

Table 1: Summary of the Literature Review

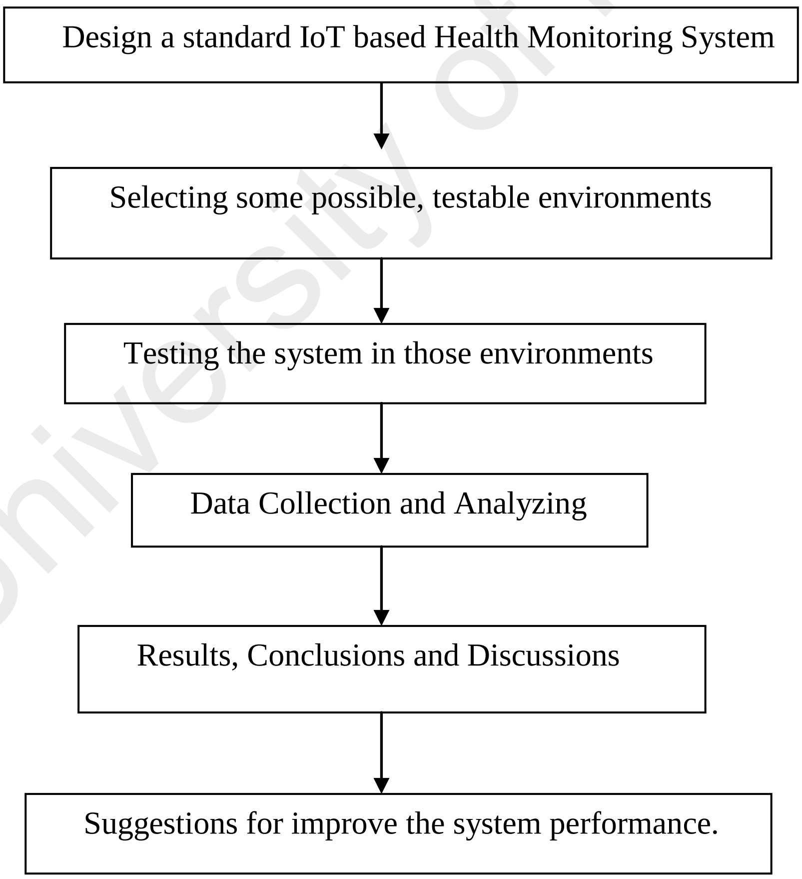
2.5. Conclusion.

Though the need for IoT based healthcare systems are increasing day-by-day to enhance the availability of healthcare to all human kind beyond the limits, still there are so many people who live under the poverty line, don't have proper access to go to hospitals for their medical checkups. With the help of latest advancements in medical industry with IoT concepts, small wireless monitoring devices can fulfill this gap and make it possible to monitor those poor patients remotely without coming to the hospitals by providing better quality of healthcare service and reduce the cost for it. Apart from monitoring the patients remotely and providing a better homecare instead of high priced clinical care, an efficient IoT based healthcare system should provide prevention alerts from possible chronic disease making factors such as living an unhealthy life, eating unhealthy foods, smoking and drinking alcohols too much and so on. These IoT based healthcare services ensure the personalization of each patient by maintaining their digital identities. Powerful IoT based systems are useful to monitor, analyze and store large amount of patients' data easily. IoT systems can provide effective healthcare services via its networked connections of multiple medical devices with cloud based storage so that the healthcare providers can observe the required medical data at the cloud storage through simply accessing their smart devices from anywhere around the world and provide fast and efficient medical supports.

CHAPTER 3: RESEARCH METHODOLOGY

In this section, first of all components are going to be selected to design a standard Remote Monitoring System based on the Literature Review. Expensive and high accuracy components are not chosen but components which are affordable and available free sources are used here since this entire project was self funded and designing a simple prototype is enough to move forward in this project. Once suitable components are chosen, connections and installation procedures with required coding are carried out. Then total expenditure, complete experimental procedures and flowchart are provided below.

3.1 Steps to be carried out in Methodology.



3.2 Components' Selection.

3.2.1 Platform Board Selection.

As the primary step, selecting a proper hardware platform is important on which the entire circuit is going to be built based on personal experience and to save a lot of time and money while designing it. Therefore for this project 'Solder less Breadboard' was chosen because it is very easy to build up a circuit on it with less effort, time and money consumptions comparing to designing a conventional PCB.

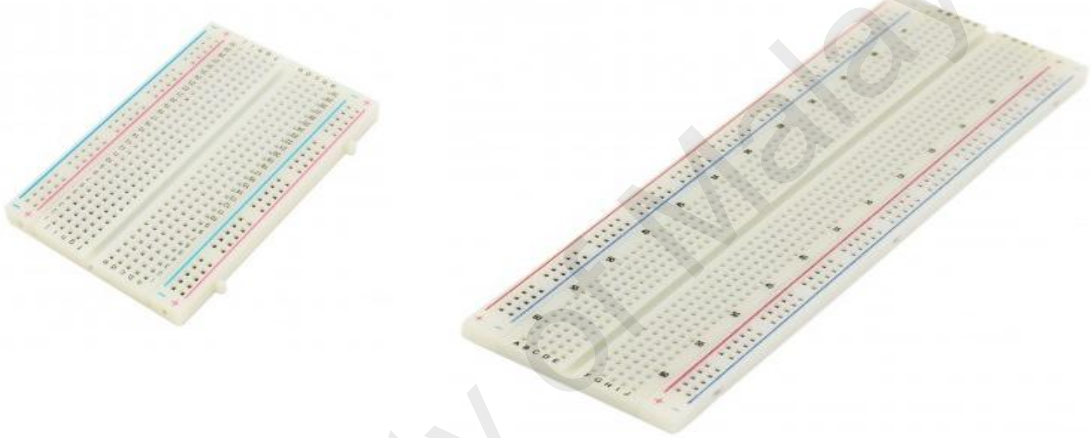


Figure 1: Left Side shows a Solder less Breadboard (Half Size) 400 tie point & Right Side shows a Solder less Breadboard (Full Size) 830 tie point

3.2.2 Processing Unit Selection

Rather than getting all the tiny electronic parts one by one and connecting them together to make a Processor Circuit such as micro controllers, resistors, capacitors and other related components. Choosing an Arduino Board which is a pre built circuit board as the Processor to this project helps in several ways. For example; Arduino is an electronic platform which provides open source software which also known as Arduino IDE and easy to use microcontroller inbuilt integrated hardware circuit with other necessary electronic components. This program runs in Mac, Windows and Linux Operating Systems and the platform is an easy tool using basic version of C++ language for easy coding and quick prototyping, therefore those who have lack of electronic and programming skills have tremendous advantage from this both hardware and software platforms. Various types of inbuilt libraries are available to assist the users for multiple functions. Several electronic components such as modules, sensors, buttons, LED's, motors and so on; can interact with this inexpensive pre assembled circuit board.

According to Elprocus (2019), there are numerous other microcontroller integrated platform boards available in the market which is similar to this Arduino such as; Netmedia's BX-24, Parallax Basic Stamp, MIT's Handyboard, Phidget and many more. But among these platforms Arduino has some major advantages over the other platforms which are:

- Inexpensive
- Cross-platform
- Simple, clear programming environment
- Open source and extensible software and hardware

To get started with Arduino, selecting the most appropriate type of board based on the functions of this prototype is vital. Some of the board types are displayed below:

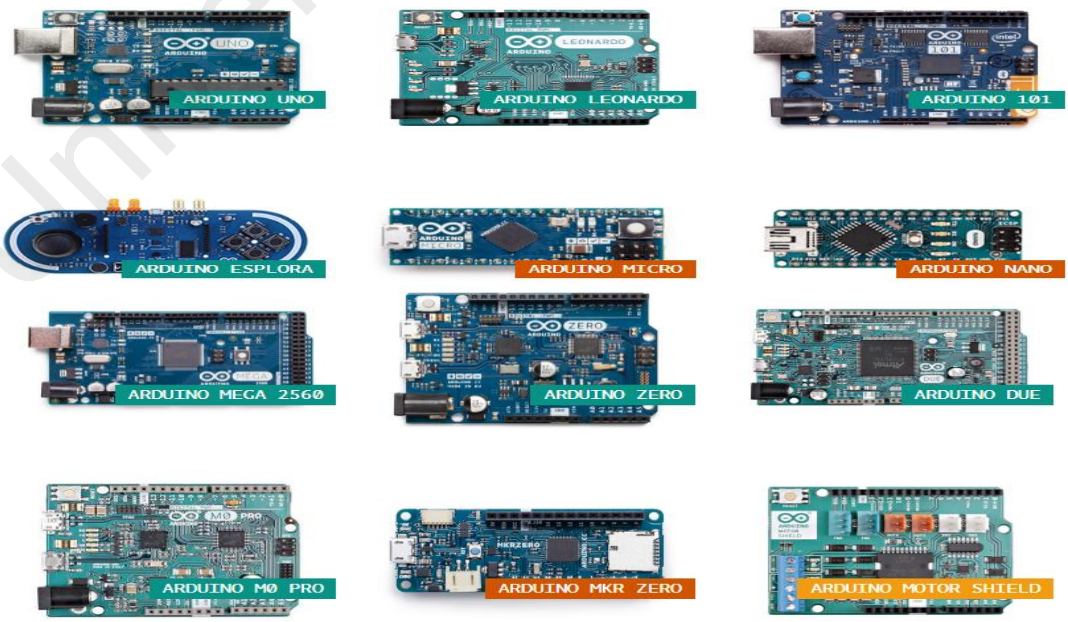


Figure 2: Types of Arduino Boards. Pictures taken from Arduino (2019).

Among these various types of Arduino Boards; ***OTA WeMos D1 CH340 WiFi Development Board (ESP8266 ESP-12E For Arduino BRD31)*** was selected based on the key functions of the prototype. Since this project is based on IoT (Internet of Things), this type of more powerful Arduino board with WiFi support was selected. This Espressif platform based on the ESP-8266 is having the pins of the Arduino Uno which is almost compatible and can be used along with most modules which are designed for typical Arduino Uno.

Figure 3: OTA WeMos D1 CH340 WiFi Development Board (ESP8266 ESP-12E for Arduino BRD31). Picture taken from www.faranux.com

As stated in Faranux (2019), the features of this board are:

- Integrated 5V; 1A switching power supply (maximum voltage 24V)
- Based on ESP-8266EX
- Arduino compatible, using IDE Arduino to program
- 11 x I/O x pin (Input/Output)
- 1 x ADC x pin (input range 0-3.3V)
- Support OTA wireless upload
- Parameters:CPU 80 Mhz,Speed of 115200
- Need to setup CH340G (USB to UART IC) driver, if first time used in a computer.

3.2.3 Getting Started with OTA WeMos D1 WiFi Development Board

This is an ESP8266 – 12 based WiFi enabled microprocessor unit on an Arduino Uno footprint. This means in most cases this board looks and functions like an Arduino Uno board. Almost most of the electronic components that are manufactured for Arduino platform will work on this board with the added advantage of built in WiFi.

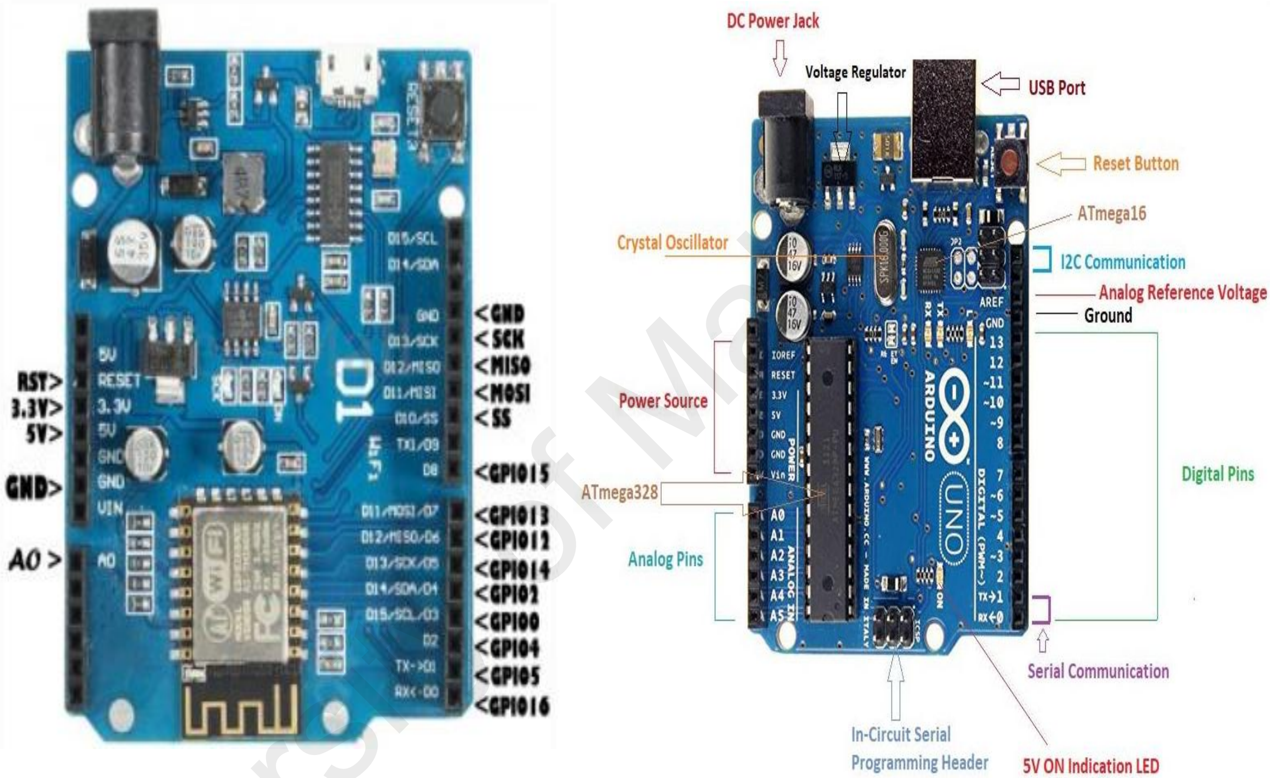


Figure 4: At left the Interface of the OTA WeMos D1 WiFi development board and at right the interface of Arduino Uno board. Taken from www.faranux.com

Even though Arduino Uno and WeMos-D1 are most similar to each other, there are a few differences in their pin alignments. The above illustrations show the mapping of the pins in both boards. Differences in the pin alignments are shown in the below table. Therefore coding written for Uno board need to be modified according to WeMos-D1 pin alignments in order to run successfully.

Arduino-UNO			WeMos-D1R2	
SCL	I2C: SCL	→	GPIO05	I2C: SCL
SDA	I2C: SDA	→	GPIO04	I2C: SDA
AREF		→		
GND		→	GND	
GPIO13	SPI: SCK	→	GPIO14	SCK
GPIO12	SPI: MISO	→	GPIO12	MISO
GPIO11	SPI: MOSI	→	GPIO13	MOSI
GPIO10	SPI: SS	→	GPIO15	SS
GPIO9		→	GPIO13	

Arduino-UNO			WeMos-D1R2	
GPIO8		→	GPIO12	
GPIO7		→	GPIO14	
GPIO6		→	GPIO2	
GPIO5		→	GPIO0	
GPIO4		→	GPIO04	
GPIO3		→	GPIO05	
GPIO2		→	GPIO16	
GPIO1	TX	→	GPIO01	TX0
GPIO0	RX	→	GPIO03	RX0

Figure 5: Different Pin Alignments in Both Boards. Table taken from www.faranux.com

3.2.4 Solving the Single ADC Port Problem in WeMos D1 Board:

Though this OTA WeMos WiFi development Board reduces the time and effort from setting up a WiFi module to an Arduino Uno board; the main inconvenience of this WeMos-D1 board is it has only one Analog Signal Port where Arduino Uno has 6 ADC Ports (A0 – A5).

But for this IoT Project, two Analog Sensors are used for getting signals from human body (Temperature Sensor and Pulse Sensor) and in future more sensors can be added to this type of prototype. Therefore a way must be found out to connect these Analog Sensors to the WeMos-D1Board.

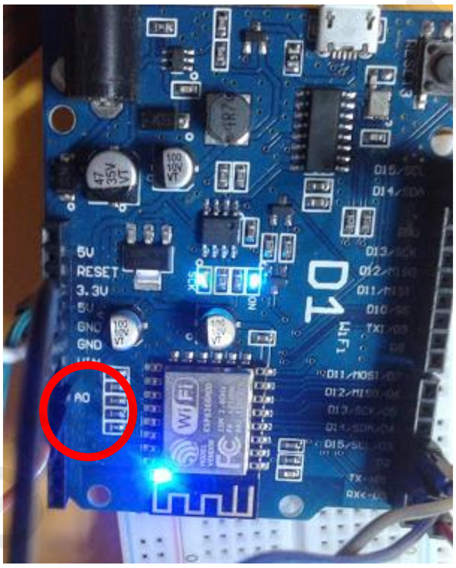


Figure 6: Only 1 Analog Signal Pin

HCF4051 - Single 8-Channel Analog Multiplexer:

A multiplexer (mux) is an electronic component which combines multiple Analog or Digital input signals and sends them to a single output line which may be from multiple Analog to single Digital or multiple Digital to single Analog. Though there are a wide range of multiplexers available; according to Github (2019), HCF4051 multiplexer, is selected to interact with the ESP8266 module since its less complexity. The diagram of the mux is given below.

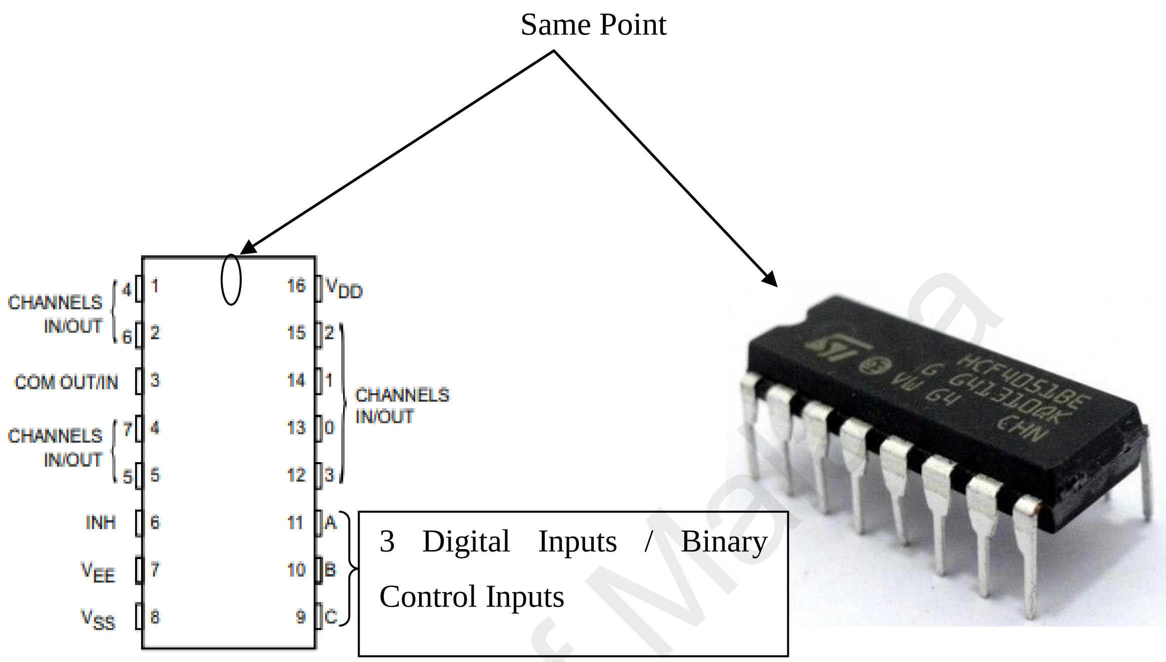


Figure 7: HCF4051 Multiplexer

According to the data sheet of this multiplexer (STMicroelectronics. 2019), it's a single 8-channel multiplexer, powered by positive supply voltage (V_{DD}) = 3.3 Voltage and having three binary control inputs (digital inputs) - A, B, and C (as showed in the diagram above). Description for the Pin Alignments for HCF4051 is given below:

Pin no.	Symbol	Name and function
11, 10, 9	A, B, C	Binary control inputs
6	INH	Inhibit inputs
13, 14, 15, 12, 1, 5, 2, 4	0 to 7 channel IN/OUT	Independent inputs/outputs
3	COM OUT/IN	Common output/input
7	V_{EE}	Supply voltage
8	V_{SS}	Negative supply voltage
16	V_{DD}	Positive supply voltage

Table 2: Pin Alignment of the HCF4051 mux. Table taken from STMicroelectronics (2019).

Truth Table of the HCF4051 Multiplexer:

The below truth table indicates on which conditions in 3 digital input (A, B and C), the In / Out Channels Gets 'ON'. Since only two analog sensors are used in this project, the first two In / Out (Channel No: 0 and 1) are selected. As this below table shows, In / Out Channel No: 0 gets 'ON' when digital inputs A, B and C are all closed (0, 0, 0) respectively and In / Out Channel No: 1 gets 'ON' when digital inputs B and C are closed and A is open which means (0, 0, 1) respectively.

Input states				"ON" channel (S)
Inhibit	C	B	A	
0	0	0	0	0
0	0	0	1	1
0	0	1	0	2
0	0	1	1	3
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
0	1	1	1	7
1	X	X	X	None

Table 3: Truth table of HCF4651 mux. Table taken from STMicroelectronics (2019).

Functional Diagram of the HCF4051 Multiplexer:

This diagram shows the pin numbers of the in / out channels according to the above truth table respectively. Therefore two analog sensors of this project need to be connected with pin no: 13 and 14 respectively because they represent in / out channel 0 and 1.

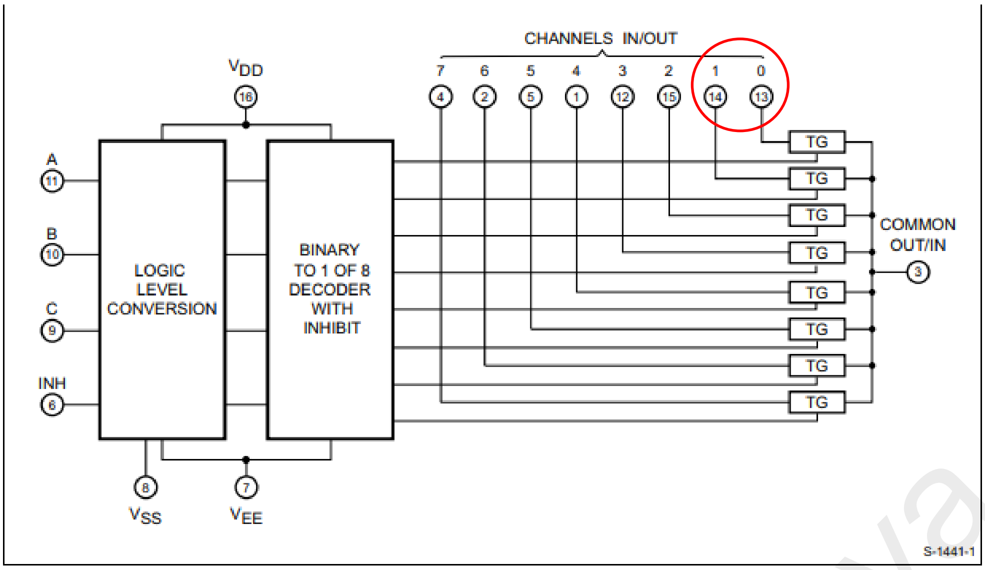


Figure 8: Functional diagram of HCF4651 mux. Table taken from STMicroelectronics (2019).

Connection Diagram between the ESP8266 Wi-Fi Module and HCF4051 Multiplexer:

The connections between these two components are based on the datasheet of HCF4051 (STMicroelectronics. 2019), and specifications of ESP8266 Github (2019).

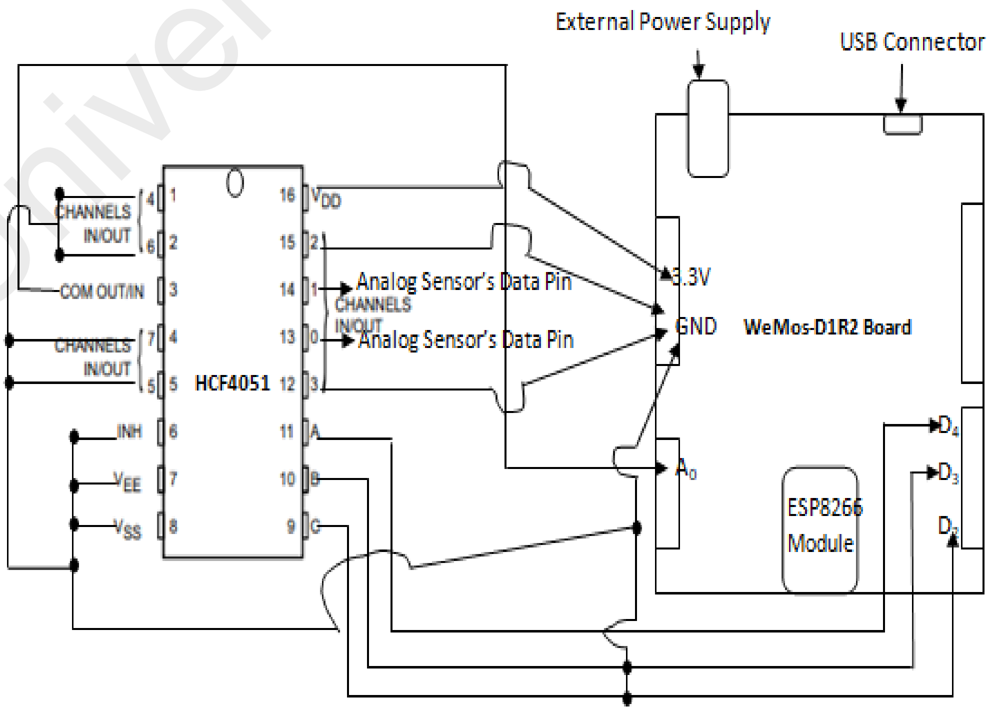


Figure 2: Connections between Arduino WeMos-D1R2 board and the HCF4051 Mux

V_{DD} is connected to 3.3 V to power up the multiplexer. The 0^{th} and the 1^{st} Channel In/Out Pins are connected to 1^{st} and the 2^{nd} Analog Sensors' Data Pins respectively (in this project Pulse sensor and Temperature sensor). In future if more sensors used they can be connected in the rest of the Channel In/Out pins (2, 3, 4, 5, 6 and 7). Therefore totally 8 analog sensors can be used via this multiplexer. Then the digital inputs (A, B and C) need to be connected to D_4, D_3 and D_2 respectively. But in this project only two Analog sensors are used and for their functions Digital Pin A only is enough (according to the above truth table) and the rest two Digital Pins (B and C) are not needed. Therefore they are connected to Ground (GND). In future, these Digitals Pins also can be used if the prototype is upgraded with more Analog Sensors (according to the above truth table). Likewise INH, V_{EE} and V_{SS} pins and the unused Channel In/Out pins (2, 3, 4, 5.6, and 7) are connected to GND as well. Then COM OUT/IN Pin is connected to A_0 terminal in the WeMos-D1R2 board. Therefore the Analog Data collected via the both Analog Sensors can reach the WeMos-D1R2 board through this pathway for processing.

3.2.5 Connections of Two Analog Sensors:

Temperature Sensor Module - LM35D:

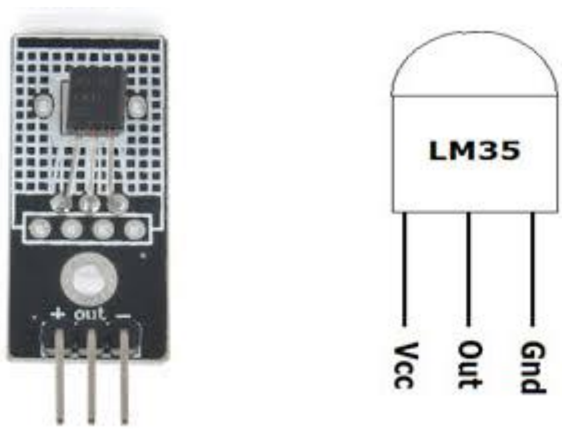


Figure 10: Top View of LM35D

According to QQ Online Trading (2019), the LM35D Linear Temperature Sensor module is based on the semiconductor LM35. The LM35 Linear Temperature Sensor module can be used to detect ambient air temperature. Its sensitivity is equal to 10mV per degree Celsius (10mV/℃). The output voltage which sends by the sensor as the output is proportional to the temperature felt by the sensor. This sensor is generally used in thermocouples, platinum resistance, thermal resistance and temperature semiconductor chips, which commonly used in high-temperature measurement thermocouples.

Even though this type of sensor is not suitable for measuring human body temperature (only suitable for measuring environmental temperature) and available most suitable human body temperatures are very expensive such as MAX30205 is around RM 200; this LM35D sensor was selected since its low cost and can use for getting rough value of human body temperature and this project is only focusing to build a prototype. Due to this sensor's less capacity for much accurate values, noise levels will be high and since the readings of this sensor are directly from the raw values, a suitable algorithm need to be given in order to get the real temperature value.

Features of LM35D Temperature Sensor:

- Based on the semiconductor LM35 temperature sensor
- Can be used to detect ambient air temperature
- Calibrated directly in ° Celsius (Centigrade)
- Linear + 10 mV/°C Scale Factor
- 0.5°C Ensure accuracy (at +25°C)

Specifications of LM35D Temperature Sensor:

- Operates voltage: DC 4V to 30V
- Current: <60uA
- Output signal type: Analog

- Sensitivity: 10mV per degree Celsius
- Functional range: 0°C to 100°C

Connection Method of LM35D Temperature Sensor:

- V_{CC} is connected to 5V of the WeMos-D1R2 board
- GND is connected to GND of the WeMos-D1R2 board
- OUT is connected to the CHANNELS IN/OUT (pin. no: 13) of the multiplexer (Refer the connection diagram of HCF4051 mux).

Heart Rate OR Pulse Sensor Module:

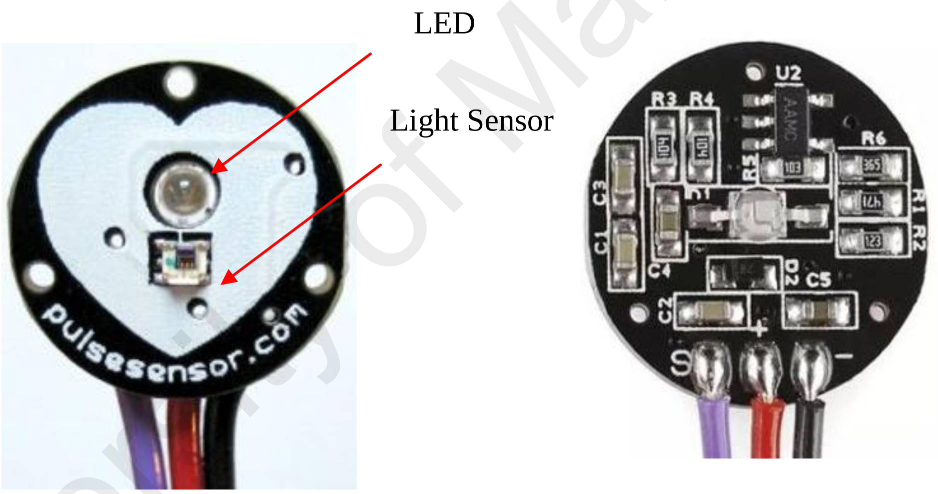


Figure 11: Front and Back Side Views of the Pulse

As Components101 (2019) states that, this type of pulse sensor is not Medically or FDA approved for accurate Heart Rate Monitoring (Pulse Monitoring) since its measuring mechanism provides a less accuracy of the heart rate.

As the images show, a LED is placed along with an ambient light sensor at the front side of the sensor. At the back side of the sensor have the circuits which are responsible for signal amplifications and noise cancelations. The LED needs to be placed directly over a vein in the human body (finger tip or ear tip). The heart pumps blood through the veins in each heart beats; so monitoring the number of blood flows in the veins in order to find the number of heart beats is the whole idea behind this sensor. When the LED sends the light through the vein, it travels through the vein and gets reflected normally. When there is blood, pushed by each heart beats, the amount of light which gets reflected will increase (reflected more from the blood). This reflected LED light can be detected by the ambient light sensor which is situated next to the LED light and this minor change in the amount of received lights between the empty vein and blood filled vein is analyzed over time to determine the heart beat.

All the vital signs of human (heart rate, body temperature, respiration rate and others) need to be measured invasively (involving the instruments into the body) rather than non-invasively because that's how a very accurate value can be obtained. But due to some medical reasons, generally non-invasive methods are used (reduce the pain while inserting the measuring probes inside the patients and reduce the time and cost while doing this procedures are some main reasons). Therefore this type of non-invasive pulse sensors can be used for only to design a prototype based on less accuracy values and projects similar to this and it is cheap compare to a much advanced and accurate sensors such as SparkFun Single Lead Heart Rate Sensor, Polar H-10 Heart Rate Sensor, etc… which may costs around RM 90 – RM 360. And further, due to this sensor's less capacity for much accurate values, noise levels will be high and since the reading of this sensor is directly from the raw values, a suitable algorithm need to be set to read then actual heart rate.

Specifications of Pulse Sensor Module:

- Working Voltage: 3V / 5V DC
- Amplification Factor: 330
- Wavelength: 609nm

- Current Consumption: 4mA
- Inbuilt Amplification and Noise cancellation circuit.
- Diameter: 0.625”
- Thickness: 0.125” Thick

Connection Method of Pulse Sensor Module:

- ‘+’ (V_{CC}) pin of the sensor needs to be connected to 5V of the WeMos-D1R2 board.
- ‘-’ (GND) pin of the sensor needs to be connected to GND of the WeMos-D1R2 board.
- ‘S’ (Signal) pin of the sensor needs to be connected to CHANNELS IN/OUT (pin. no: 14) of the multiplexer. (Refer the connection diagram of HCF5041 mux).

Connection of both Sensors and the Multiplexer with WeMos D1R2 Board:

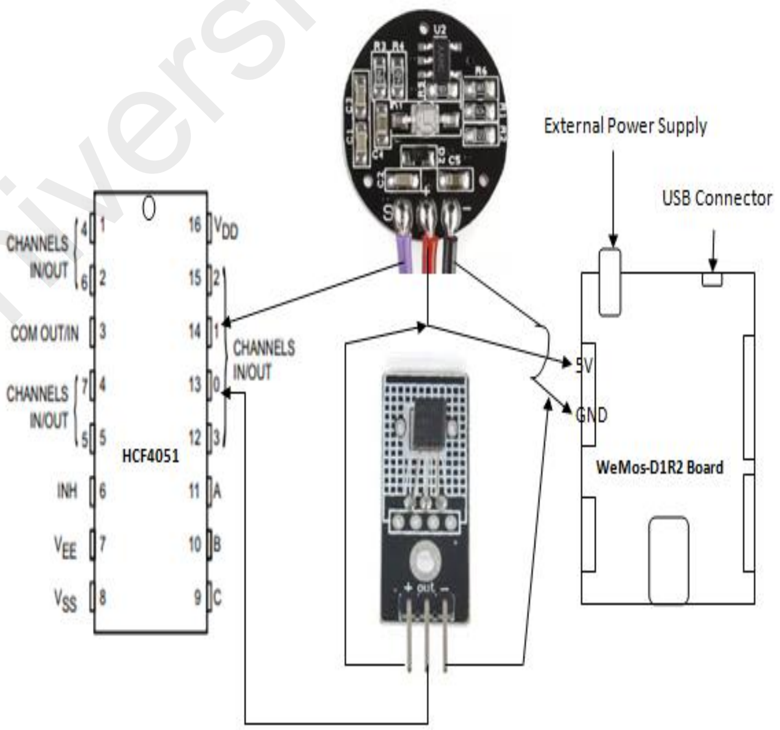


Figure 12: Connection of all components

3.2.6. Connecting 16 x 2 LCD Display with Serial LCD I2C Module (I2C LCD Screen Controller).

16x2 LCD Display:

Figure 13: 16x2 LCD

Liquid Crystal Display which uses here is having 16 x 2 characters with green backlight. This means this LCD can print 16 characters (a character means a letter or number or symbol, etc…) vertically and likewise it can print in 2 rows (horizontally).

Specifications:

- Number of Characters: 16 characters x 2 line
- Input Voltage Range: DC 4.5V to 5.5V
- Operation Voltage for LCD: 5V DC
- Backlight Forward Current: 120 mA
- Backlight Forward Voltage Range: 4.1 V to 4.3 V
- Module Size: 80.0 x 36.0 x 13.5mm
- Weight: 40g

Pin Mapping of 16x2 LCD:

No. of the Pin in LCD	Pin Name	Function
1	V_{SS}	Ground
2	V_{CC}	DC 5V power supply
3	V_{EE}	LCD contrast adjustment
4	RS	Instruction / data register selection. RS = 0 - Instruction register RS = 1- Data register
5	RW	Read / Write selection
6	E	Enable signal. Active when high-level turns to low-level
7 - 14	D0 – D7	Data input / output lines; 8-bit: D0-D7
15	A	Anode - Backlight voltage: Positive
16	K	Cathode - Backlight voltage: Negative

Table 4: Pin Mapping of 16x2 LCD

I2C LCD Screen Controller:

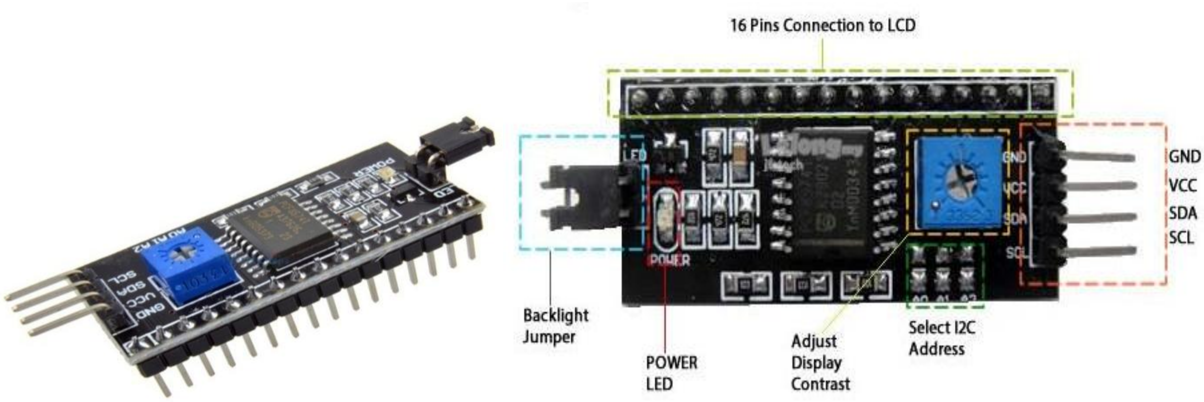


Figure 14: Front view of the I2C Module

This module has a PCF8574T semiconductor (IC) onboard which reduces the number of data pins needed to control the LCD down to 2. This excludes the power / ground pins which are also required. This I2C-Backpack featuring the PCF8574T-I2C-Driver can be soldered behind a standard 1602- or 2004-LCD - after that, only 2 I/O-Pins are required to get the LCD up and running. Additionally, an onboard potentiometer is there which helps to control the contrast as well as completely disable the backlight with a jumper.

Features of I2C LCD Controller:

- Applicable for all standard 5V 16 Characters, 2 Lines and 20 Characters, 4 Lines LCD
- PCF8574T Driver
- Potentiometer for Contrast
- Disable Backlight by jumper
- Connector Pins: SDA / SCL / VCC / GND
- Size: 40mm x 18mm

Pin Alignment of the I2C LCD Controller:

Pin No.	Symbol	Description
1	VSS	Ground(0V).
2	VDD	Power supply for logic (+5V)
3	V0	Power supply for LCD driver
4	RS	Register Select Input: "High" for Data register (for read and write) "Low" for Instruction register (for write), Busy flag, address counter (for read)
5	R/W	Read/Write signal: " High" for Read mode. "Low" for Write mode.
6	E	Enable. Start signal for data read /write.
7	DB0	Data input/output (LSB)
8	DB1	Data input/output
9	DB2	Data input/output
10	DB3	Data input/output
11	DB4	Data input/output
12	DB5	Data input/output
13	DB6	Data input/output
14	DB7	Data input/output (MSB)
15	LED(+)	Anode of LED backlight
16	LED(-)	Cathode of LED backlight

Table 5: Pin Alignments of I2C LCD Controller

Therefore these 16 pins of I2C LCD Controller are needed to be connected with those 16 pins of the 16x2 LCD respectively. (E.g.: Pin 1 of I2C connects with Pin 1 of LCD, etc…)

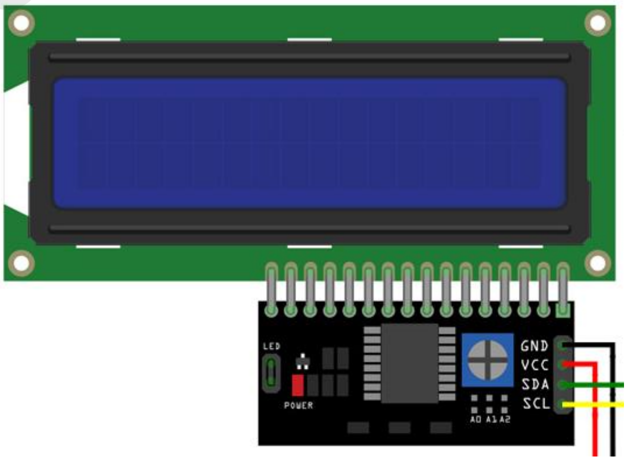


Figure 15: Connections between 16x2 LCD and I2C Module

Apart from these 16 pins explained in the above image, there are 4 more important pins available on the surface of the I2C Controller. Those pins, its functions and connection diagrams are given below:

Connection Method of I2C LCD Controller:

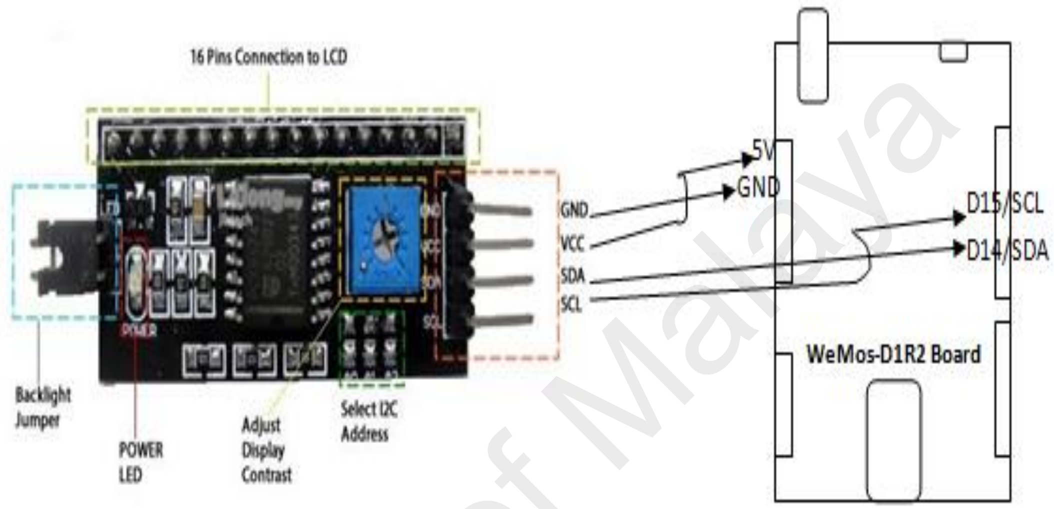


Figure 16: Connecting I2C Module with WeMos Board

Functions and Connecting Points of the Pins:

Pin Name	**Function**	**Connection Point to D1 Board**
GND	Ground	GND
VCC	Input Power Supply	5V
SDA	Serial Data Line	D14 / SDA
SCL	Serial Clock Line	D15 / SCL

Table 6: Pin Functions and Connections of I2C LCD Controller

3.3 Programming the System:

Once the hardware connections among these components are over, they must be programmed to how they should work together and give results. Arduino IDE provides its own open source software Arduino, for all its hardware components. This software consists of several libraries which can support to create the entire coding for all these hardware components and its synchronization. The entire coding and its functions is provided in **Appendix A**

3.4 Completed Prototype:

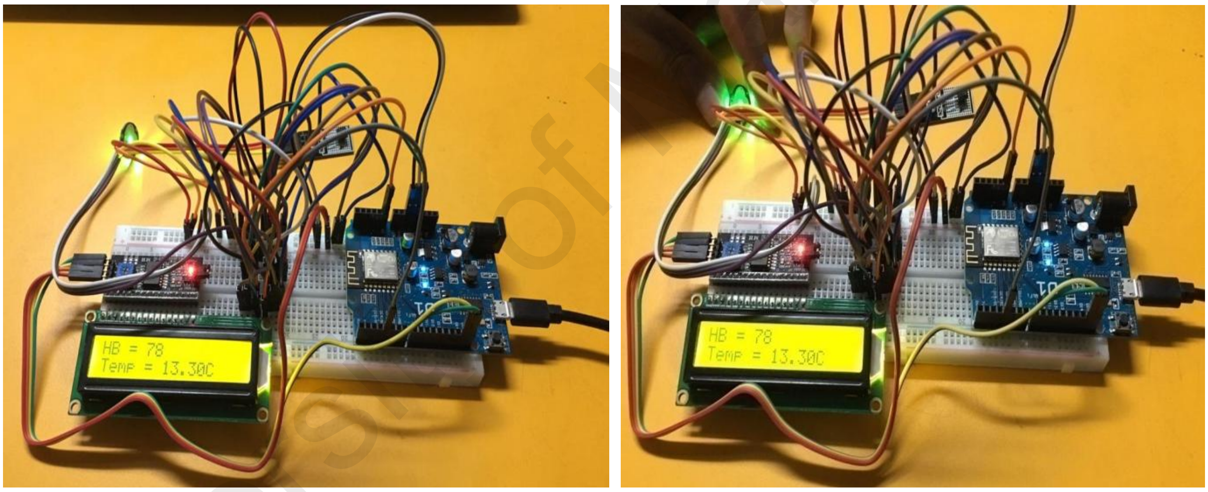


Figure 17: Images of Completed Prototype

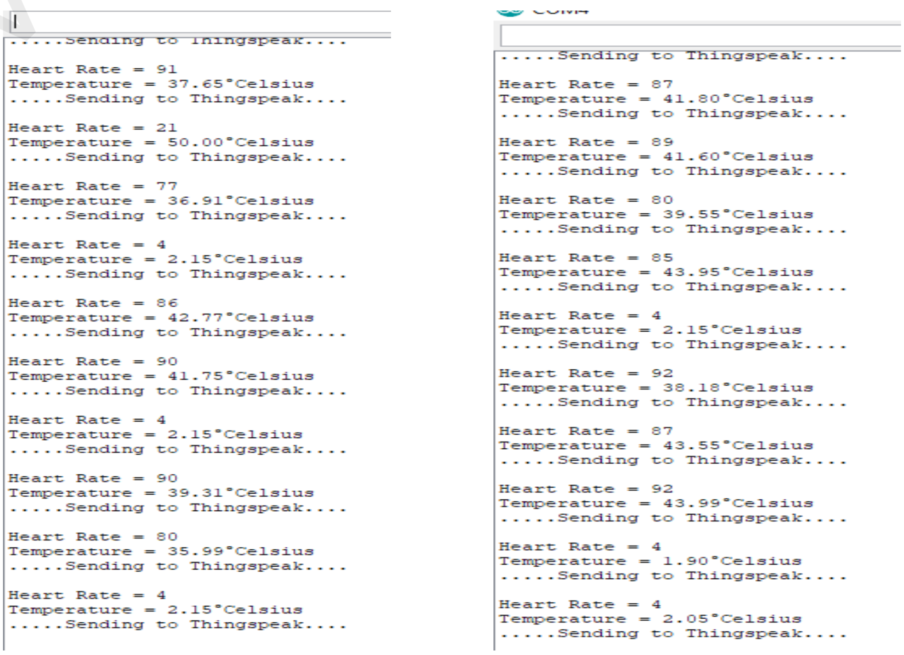


Figure 18: Data receiving images from Serial Monitor

Once the pulse sensor is kept in between the two fingers and the temperature sensor is at the palm as demonstrated in the above prototype image (or can be kept by the other hand) the LCD shows the sensor values as well as the serial monitor.

3.5 Total Expenditure for this Project:

Components	**Quantity**	**Price in Ringgit Malaysia (RM)**
Breadboard (830 Hole)	1	13.00
WeMos D1 WiFi board	1	35.00
Micro USB Male-Type A Male Cable	1	06.00
LED Lights	5	01.00
Jumper wire (Female-Male)	10	02.00
Jumper wire (Male-Male)	20	04.00
LCD Display 16x2	1	13.00
I2C LCD Controller Module	1	06.00
HCF4051 Multiplexer	1	02.50
Pulse Sensor	1	35.00
Temperature Sensor	1	10.00
Total		127.50

Table 7: Total Expenditure for the Project

Note that RM 10.00 need to be added with this amount which was spent in travelling to buy these things at **QQONLINE TRADING**; situated in *24-1-G, Jalan Landak, Pudu, 55100 Kuala Lumpur, Malaysia*.

Therefore Total Expenditure for this Project is = RM 137.50

3.6 Internet of Things Cloud Based Server Setup:

Once designing the prototype is over, the next step is to set up the IoT Cloud based server for data analysis, display and storage. Nowadays there are a lot of cloud based servers available since everything in our day to day life is being based on IoT. Amazon Web Services (AWS), Microsoft Azure, IBM Watson, Google Cloud, Oracle, Thingworx 8, Cisco, Bosch, etc… are some top most IoT Cloud based Server providers. Among these cloud servers a suitable platform needs to be selected to demonstrate this project’s concept.

ThingSpeak™ is an IoT analytical platform which provides services to collect analyze and visualize live data streams in various graphical forms to understand the results easily in the cloud. This open IoT platform with MATLAB® Analytics is mostly used for testing IoT based prototypes. The main interface of ThingSpeak™ and various data representation methods available in the platform are given below:

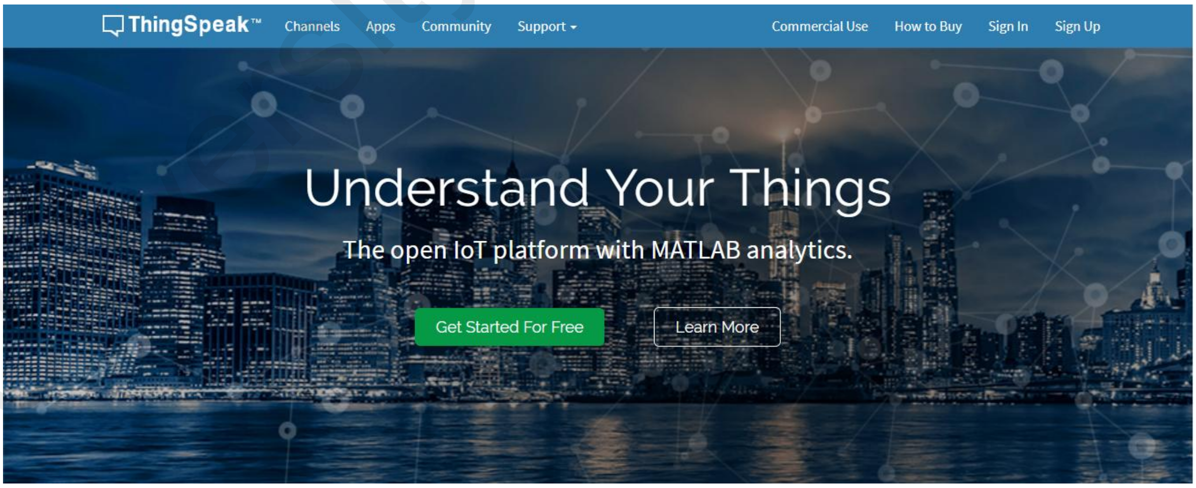


Figure 19: Interface of the ThingSpeak™ Platform

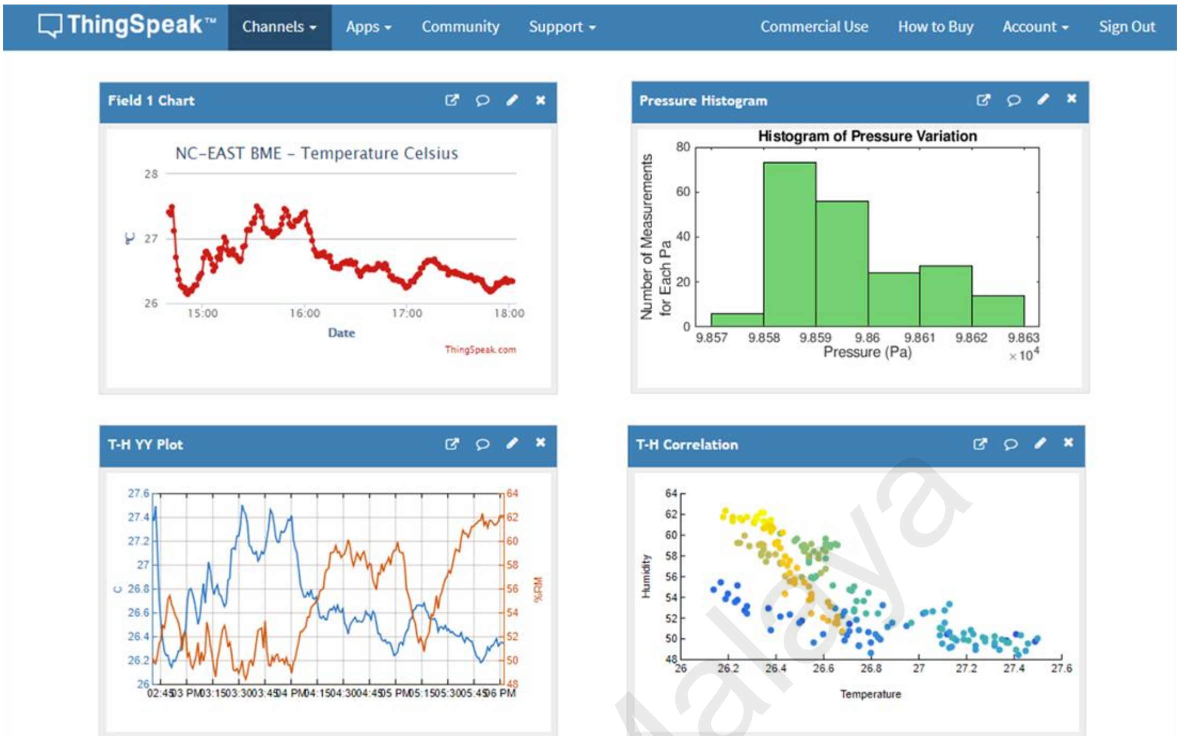


Figure 20: Different data representational methods available at ThingSpeak™ Platform

Apart from that, some key features of ThingSpeak™ are:

- Easy configuration to send data to ThingSpeak™.
- Visualize sensor data in real-time in various data representation methods.
- Data from the cloud can be obtained from third-party sources (smart phones, tabs, etc...)
- Run the IoT analytics automatically based on schedules or events.
- Prototype and build IoT systems without setting up servers or developing web software.
- Storing facility for data which can be obtained at anytime, anywhere over the internet by simply key in the username and password.

Therefore this will be a sufficient platform for this project's purpose.

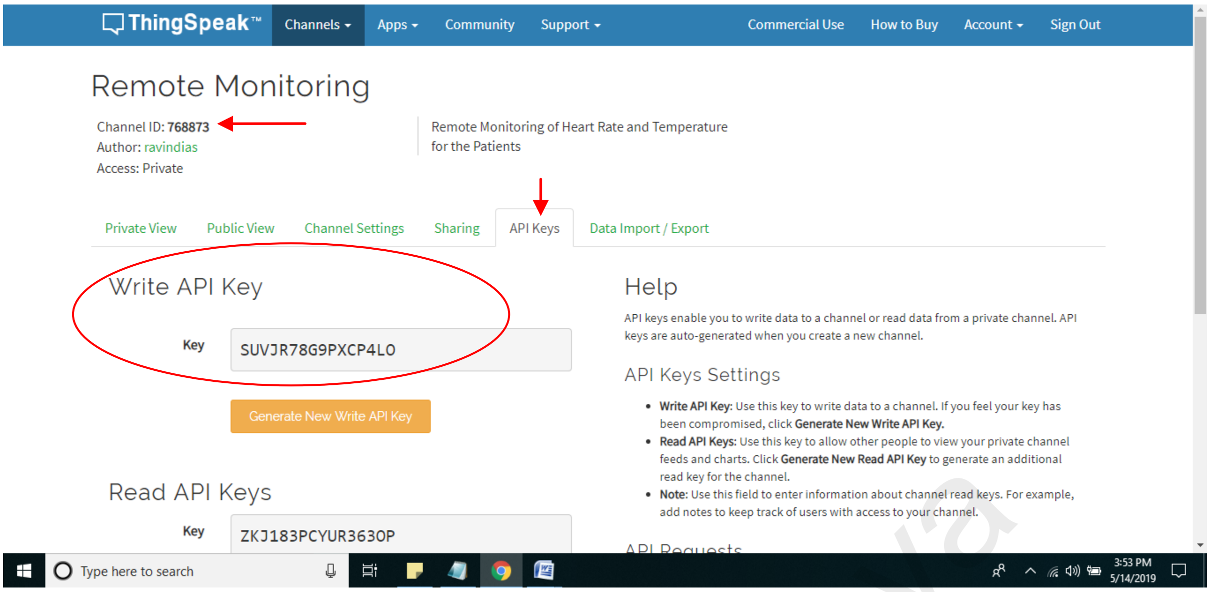


Figure 21: Important Numbers for Programming

Channel ID and Write API Key (Application Programming Interface) which acts as a communication protocol (marked in Red) are the two major key numbers need to be entered in the Arduino coding for the prototype to send the data to this particular Channel.

Apart from this PC interface, smart phone interface is also available for this ThingSpeak™ platform where all the data can be obtained more easily by simply key in the Channel ID and the API Key. Since these numbers differ from patient to patient, healthcare providers can simply key in those numbers in order to review the patients' vital sign history respectively.

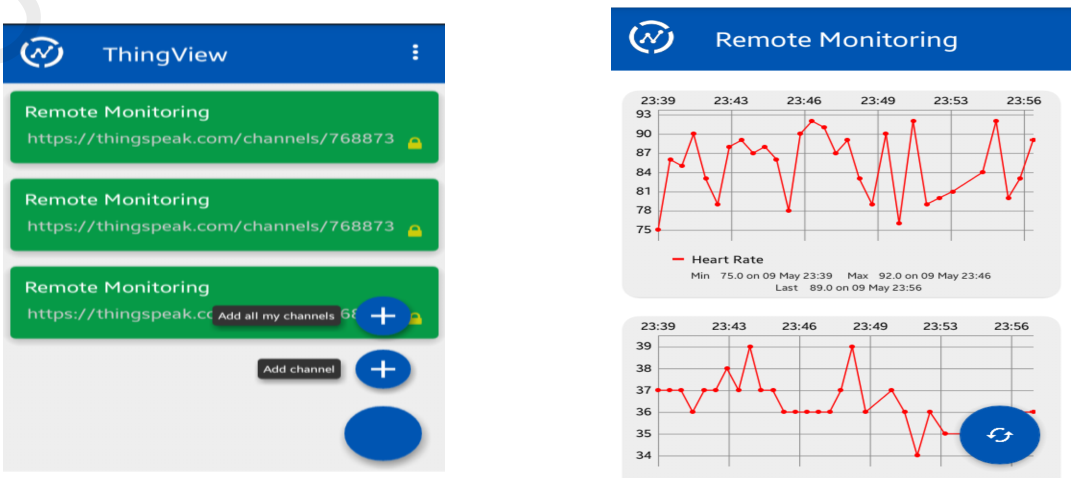


Figure 22: Smart phone view of the IoT Platform

3.7 Experiment Procedure:

Pulse sensor need to be attached with the tip of any fingers but not too tight. Likewise temperature sensor can be kept at the palm and close all the fingers tightly or kept at the other hand similar to the pulse sensor in order to give enough body temperature. Each test is carried out for not less than 15 minutes continuously at different locations.

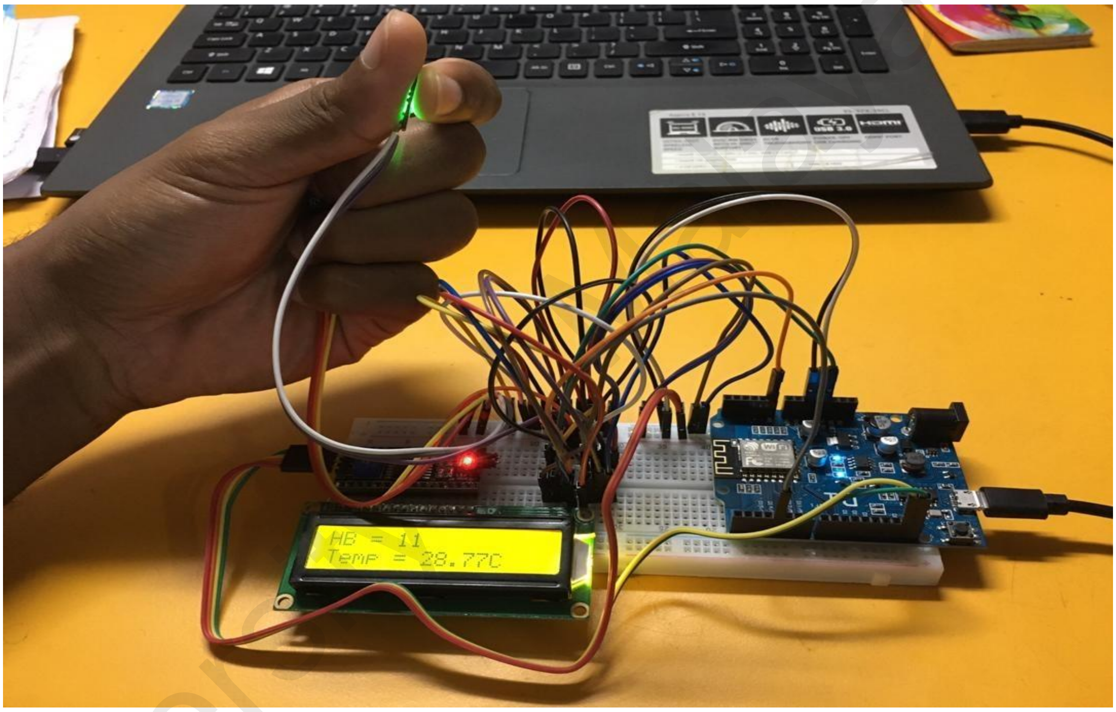


Figure 23: Demonstrating the holding technique of both sensors.

Then at the serial monitor, the system is searching for mobile network which is personal hotspot can be displayed:

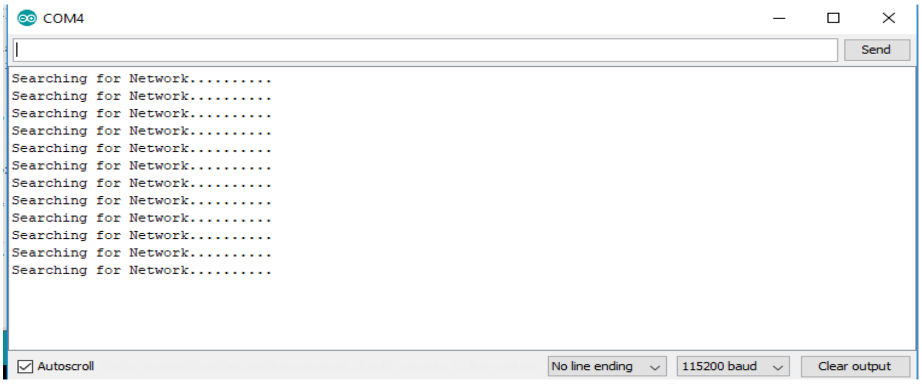


Figure 24: Searching for Defined Mobile Hotspot.

Once the mobile hotspot is turned on this system gets initiated to measure the Pulse Rate and the Body Temperature which is displaying at the serial monitor as well as at the LCD screen at the same time:

```
Trying to Connect with >>>>> AndroidAP #Gaston#
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........
Searching for Network..........

Hotspot Connected Successfully
D1 Board IP Address = 192.168.43.6
Heart Rate = 141
Temperature = 46.41°Celsius
.....Sending to Thingspeak....
```

Considered as Patient

Figure 25: Starting to get values from both sensors.

Above image shows that the system is trying to connect with the defined network which was personal hotspot in this case. Once connected it shows “Hotspot Connected Successfully”. If not connected, the system will continuously search for the defined network.

The next line shows the D1 board’s IP Address which is unique and vary from board to board. Therefore this D1 Board’s IP address can be used as Patient’s ID which also is unique to each patient to identify them. Then the readings from the Heart Rate and Temperature sensors are displayed respectively. At last these data are sent to ThingSpeak™ IoT platform.

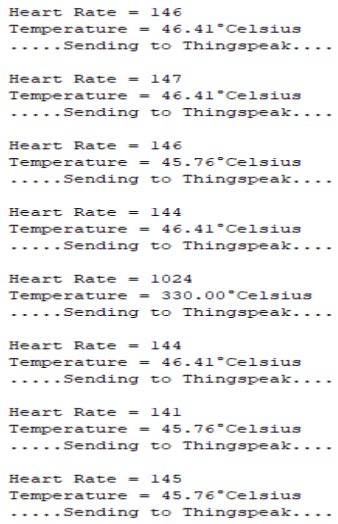

```
Heart Rate = 146
Temperature = 46.41°Celsius
.....Sending to Thingspeak....

Heart Rate = 147
Temperature = 46.41°Celsius
.....Sending to Thingspeak....

Heart Rate = 146
Temperature = 45.76°Celsius
.....Sending to Thingspeak....

Heart Rate = 144
Temperature = 46.41°Celsius
.....Sending to Thingspeak....

Heart Rate = 1024
Temperature = 330.00°Celsius
.....Sending to Thingspeak....

Heart Rate = 144
Temperature = 46.41°Celsius
.....Sending to Thingspeak....

Heart Rate = 141
Temperature = 45.76°Celsius
.....Sending to Thingspeak....

Heart Rate = 145
Temperature = 45.76°Celsius
.....Sending to Thingspeak....
```

Figure 26: Continuous data from the sensors to serial monitor

Then switching to the ThingSpeak™'s 'Private View' to check the sending data from the system to IoT platform which provides the received data into graphical forms for easy analyzing.

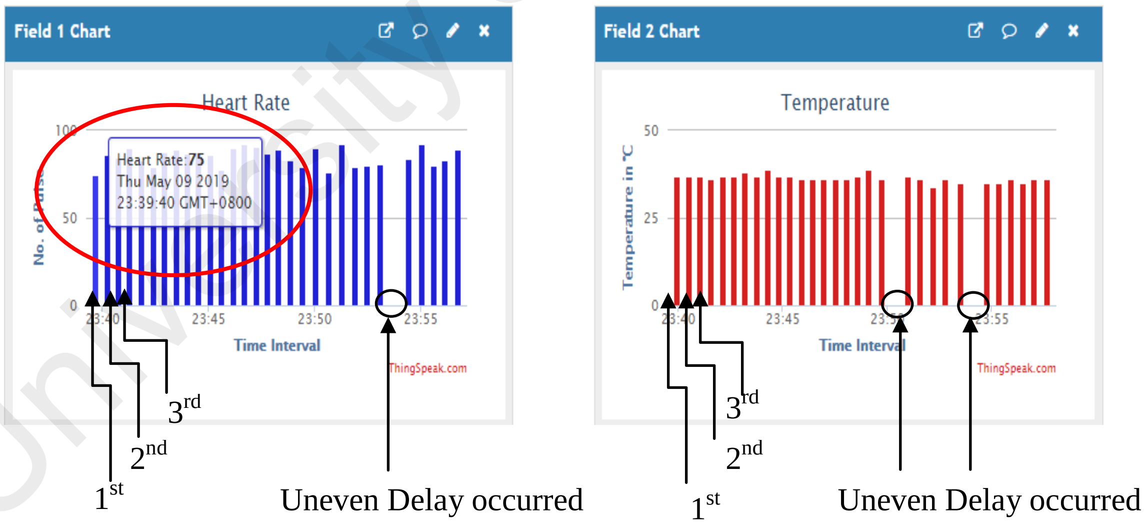
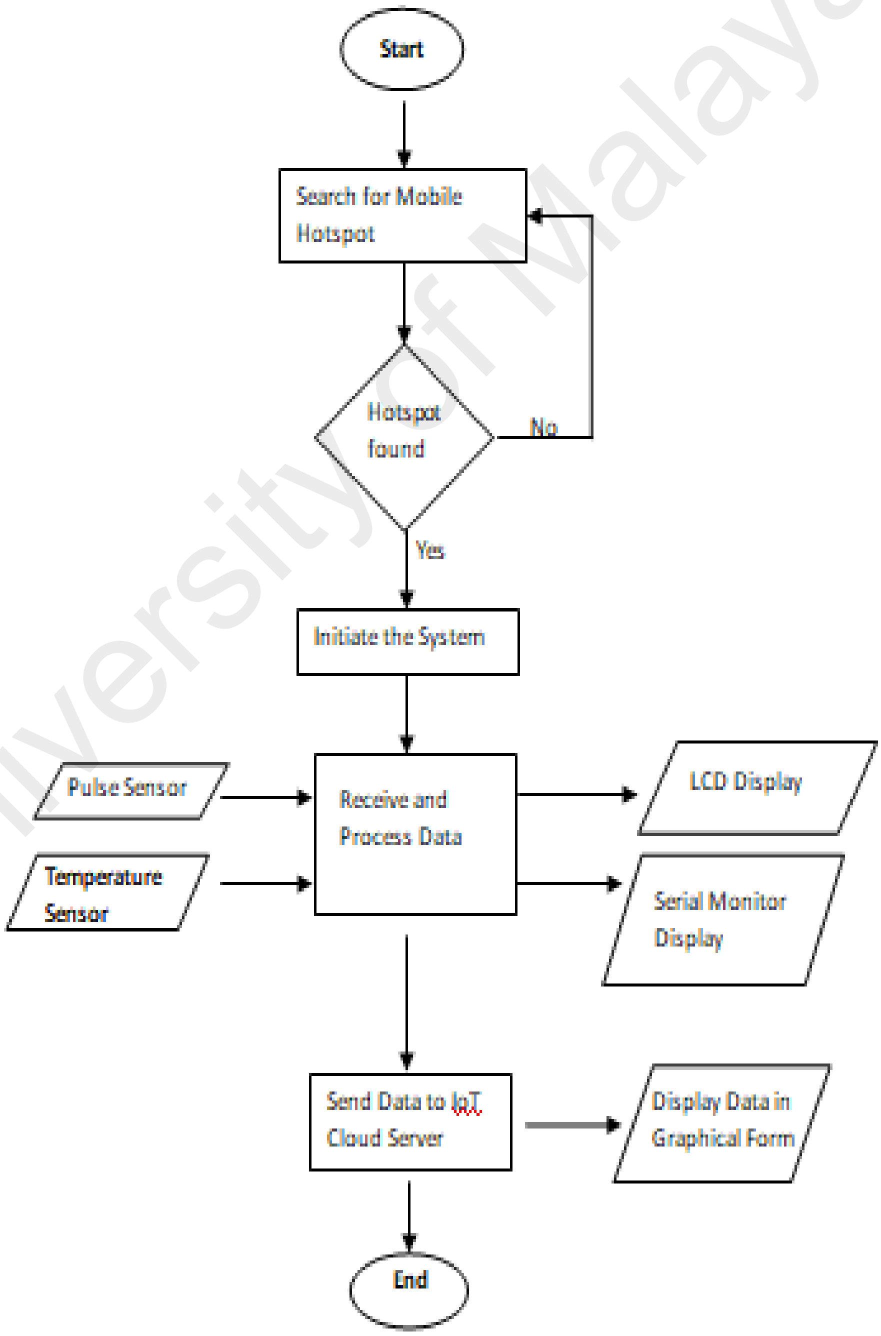


Figure 27: Information from the data and its counting method.

These each bar graphs display data from both sensors. Counting method of the data is displayed on the image. In both graphs Y-axis denote Number of pulse and body temperature respectively and X-axis shows time. To find out the details of any data, simply placing the mouse pointer on the data is enough. Above red circle in the image was a data which shows Heart Rate was 75 at the particular time on the particular date. Likewise the data for the body temperature can be obtained.

As a main objective of this project, the reasons for the delay between each data from both sensors and reasons for uneven delay occurred are going to be found out by doing tests in multiple environments.

3.8 Flow Chart for the Entire System:



3.9 Summary and Conclusion of the System:

As the above flow chart shows, at the initial stage the system searches for the defined mobile hotspot (in this case, personal hotspot is used. Likewise anyone can give their own hotspot details for their own system). Once the defined hotspot is detected and the system is connected to the hotspot, system starts gathering data from the sensors. If not the hotspot is connected or detected the system continuously searches for it.

The data gathered by the sensors are sent to the Arduino Wemos board and it displays them at the LCD Screen as well as at the Serial Monitor at the same time. LCD represents showing their own vital signs to those who (patients) wear this system and Serial Monitor is considered as monitoring facility for the healthcare provider such as nearby hospitals, doctors, nurses, relatives, etc… If the sensors readings go above or below the pre defined values (like the normal heart rate is 80-100BPM and Body Temperature is around 36℃-38℃) the healthcare provider can sent assistance as soon as possible to the respective patient.

At the same time, from the Arduino Wemos board these collected data are sent to the Thinkspeak™ IoT cloud based server for analyzing, and displaying in preferred methods such as different graphical forms. Furthermore old medical data can be stored in order to review the entire history of a patient if needed. This IoT platform can be accessed from anywhere around the world via any digital devices like computers, smart phones, tabs, etc…by simply key in the authorized username and password by the third parties who can monitor, and advice about the patient's health like getting an opinion from an oversea consultant, getting second or third opinion from a doctor who is in abroad, showing the entire medical record to a consultant who stays in abroad, etc…

Test No: 01

Location: Room No A005, KK12, University of Malaya.

Average Wi-Fi Speed:

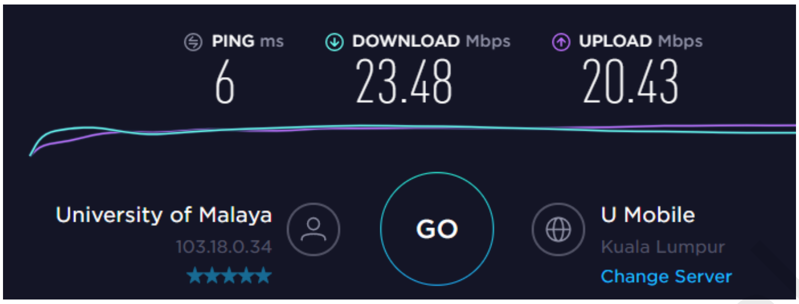


Figure 4: Average Wi-Fi Speed at the Room

Average Mobile Signal Strength during the Test:

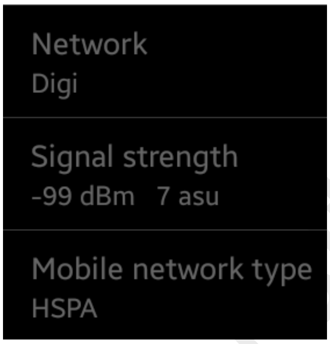


Figure 5: Average Mobile Signal Strength at the Room

Data Obtained at ThingSpeak™ IoT Platform:

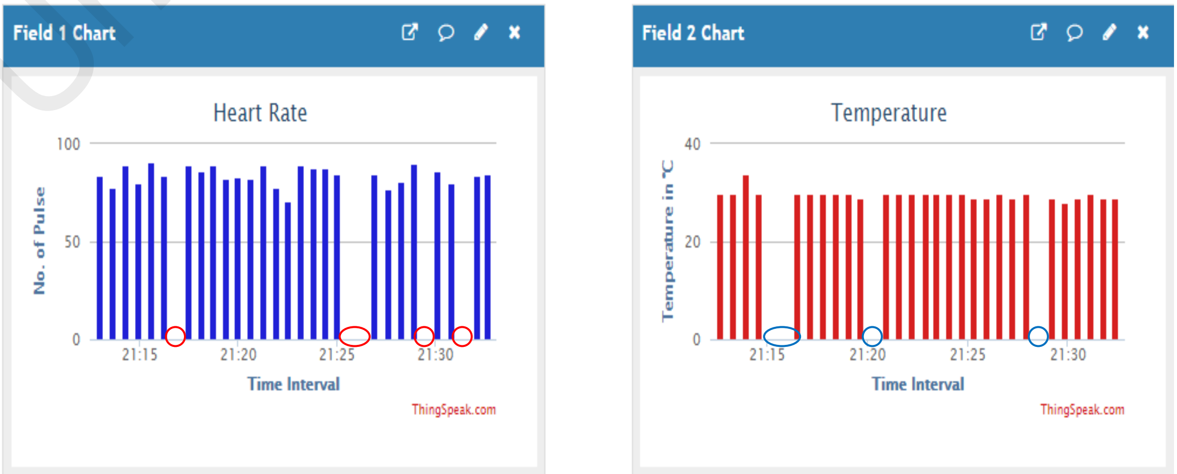


Figure 6: Data obtained at the IoT Server from Room

- Data were recorded from both sensors for same time period. E.g. Data between 21:15 – 21:30 from both sensors can be observed here.
- There were some uneven time delays occurred in both Charts which are circled in Red and Blue colors. (In Heart Rate 4 times and in Temperature 3 times).
- Values of these sensors were not stable and fluctuated for some extent. Heart Rate fluctuated from 71 BPM – 91 BPM during the focused time frame and the rest of the values fall in between these two limits. Likewise Temperature varied from 29°C - 34°C and the rest of the readings fall in between these numbers.

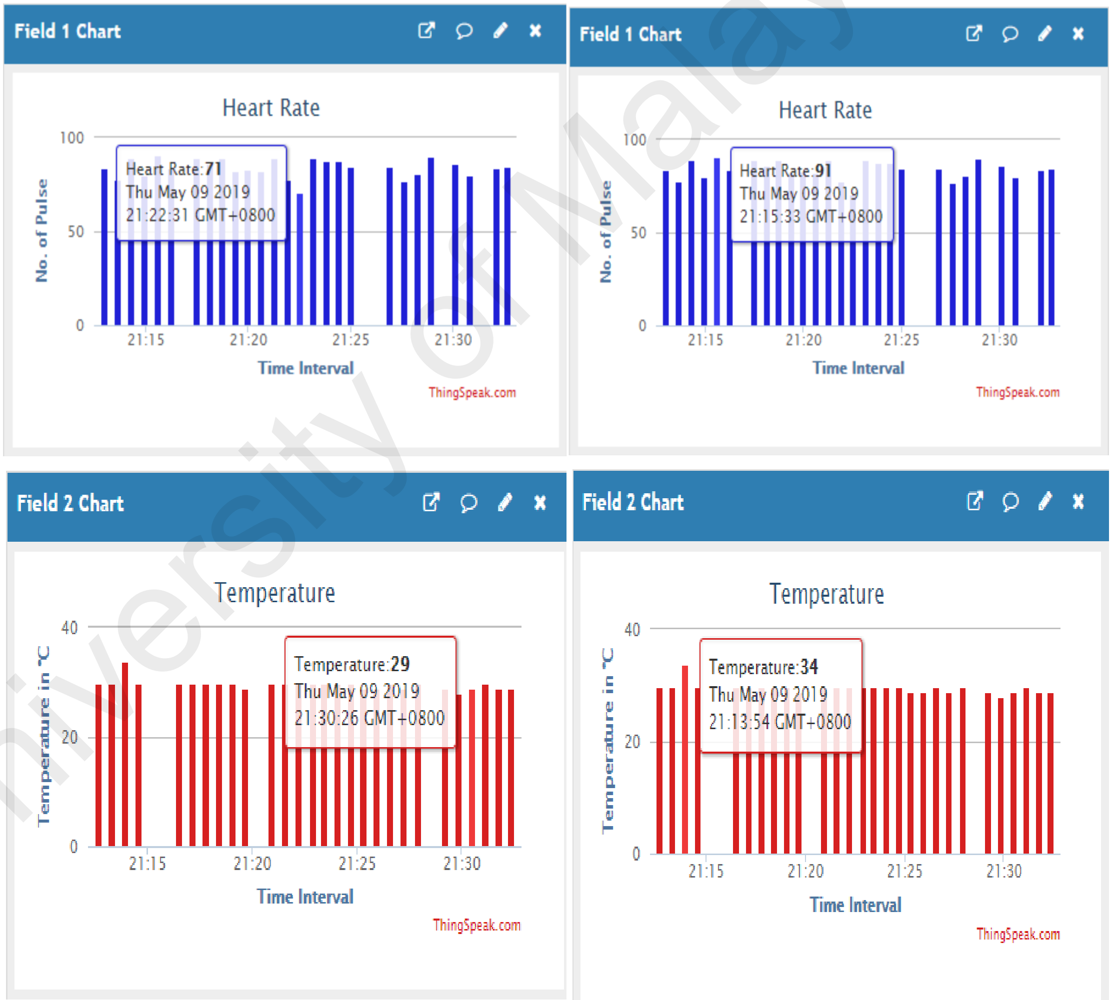


Figure 7: Pulse Sensor and Temperature data obtained at the Room

Data Analyzing:

Considering 6 consecutive data from each sensor to find out the time delay between each consecutive data. Getting 5 such data delays (samples) will be helpful to find out the average data delay in this test. Starting from data number 12 which is selected randomly and there are no uneven time delay occurred after this data in both sensors for some certain extend which provides a number of continuous data for analyzing:

Heart Rate Data:

12th data received at 21:20:35

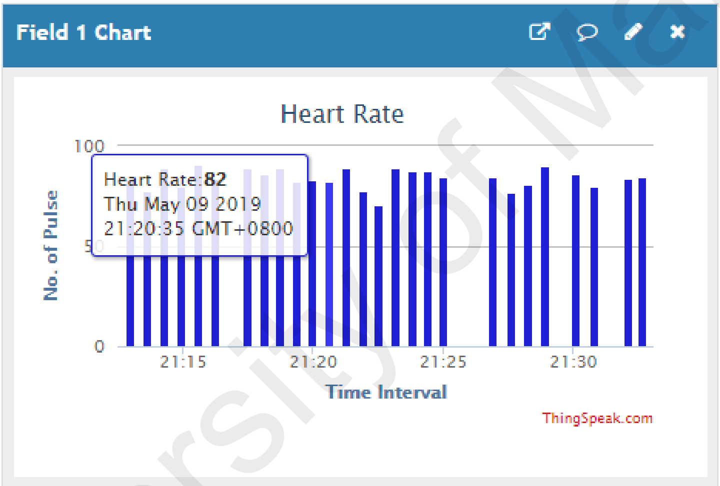


Figure 8:12th data

13th data received at 21:21:14

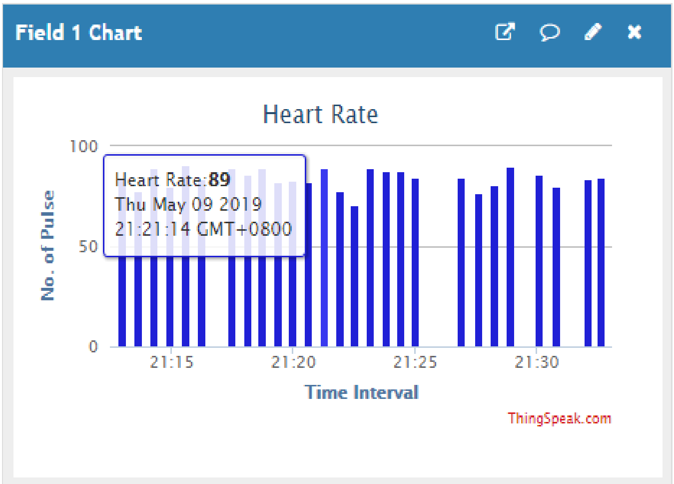


Figure 9: 13th data

14th data received at 21:21:53

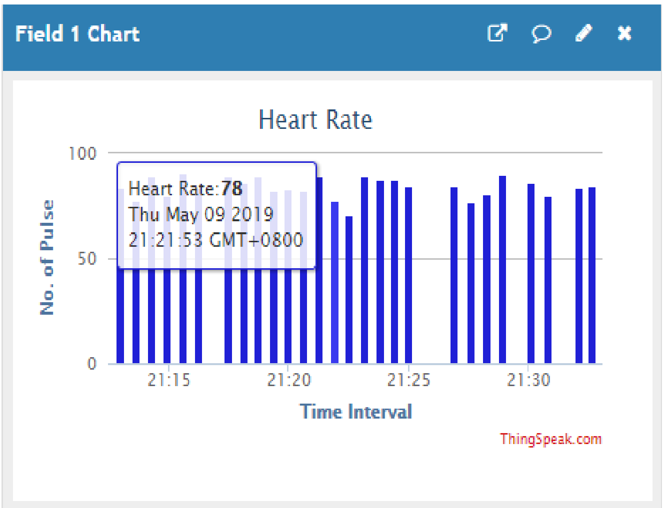


Figure 10:14th data

15th data received at 21:22:31

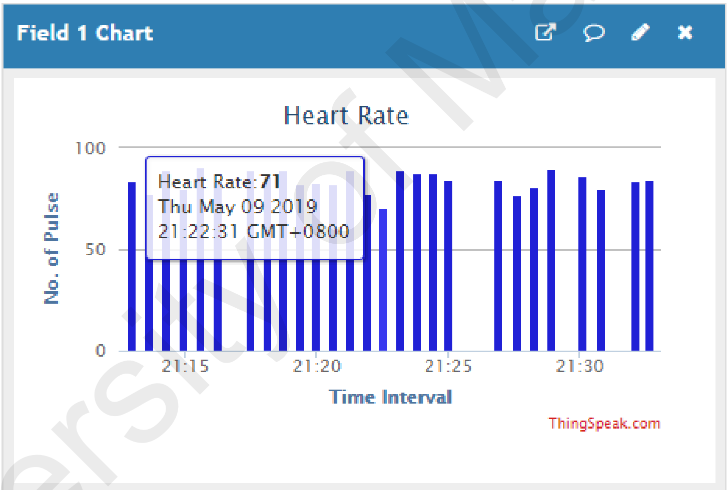


Figure 11: 15th data

16th data received at 21:23:09

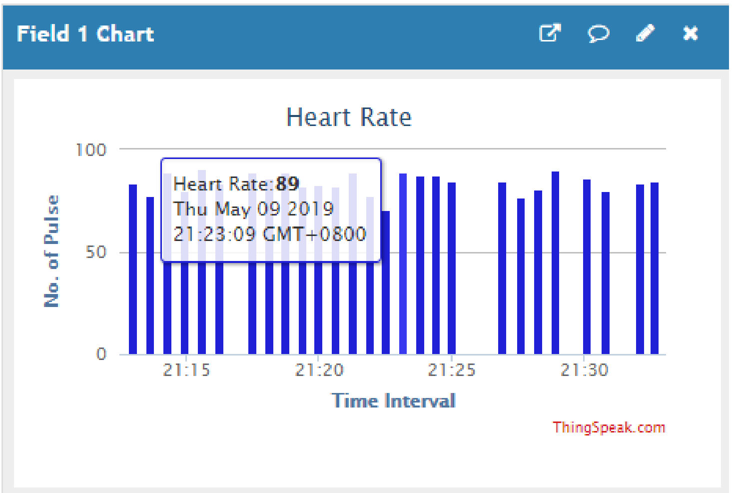


Figure 12: 16th data

17^{th} data received at 21:23:46

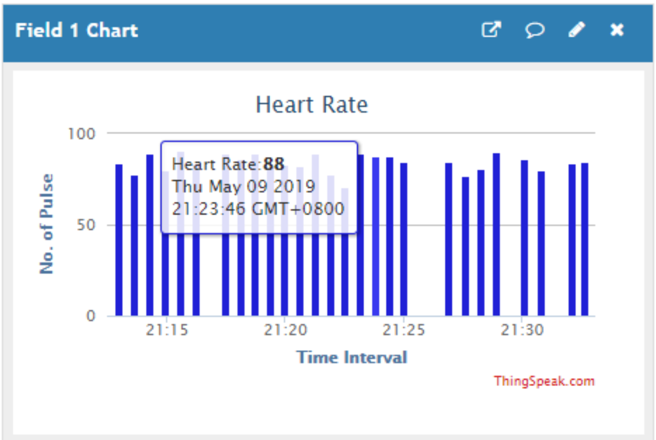


Figure 13: 17th data

Summary:

Data No.	Received Time	Delay between Each Data (in Seconds)
12^{th} Data	21:20:35	-
13^{th} Data	21:21:14	39
14^{th} Data	21:21:53	39
15^{th} Data	21:22:31	38
16^{th} Data	21:23:09	38
17^{th} Data	21:23:46	37
Average Delay in between each data		(39+39+38+38+37)/5 = 38.2

Table 8: Summary Table of Pulse Sensor at Room

Therefore in this environment the average delay between each pulse sensor data was 38.2 seconds.

Temperature data:

12th data received at 21:21:33

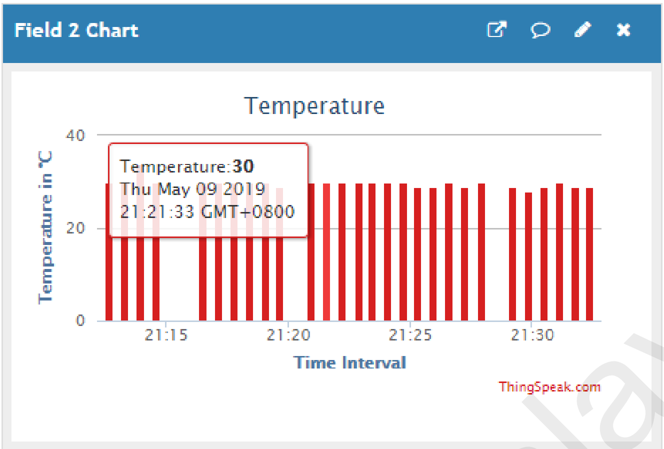


Figure 14: 12th data

13th data received at 21:22:11

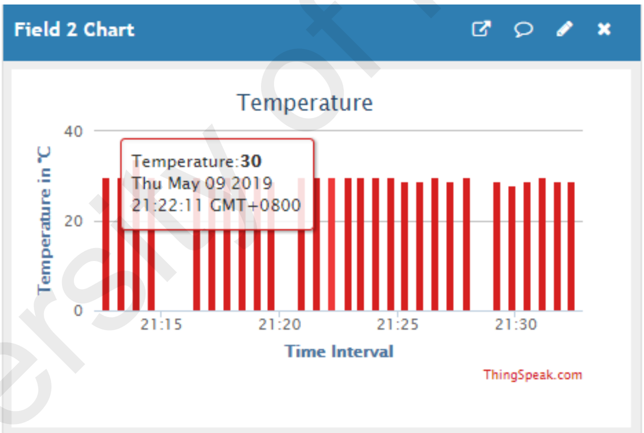


Figure 15: 13th data

14th data received at 21:22:52

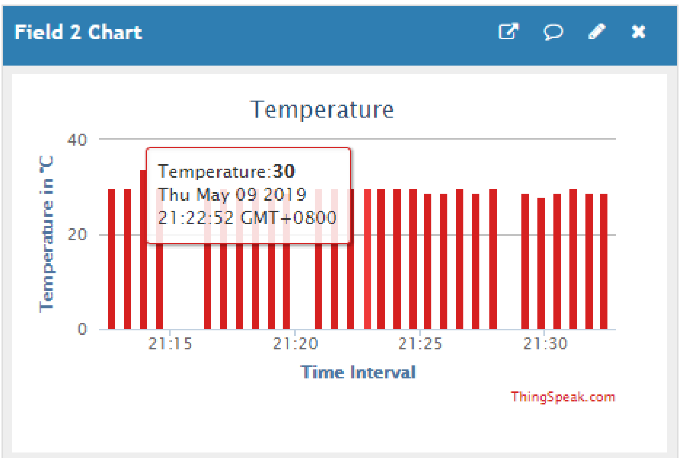


Figure 16: 14th Data

15th data received at 21:23:26

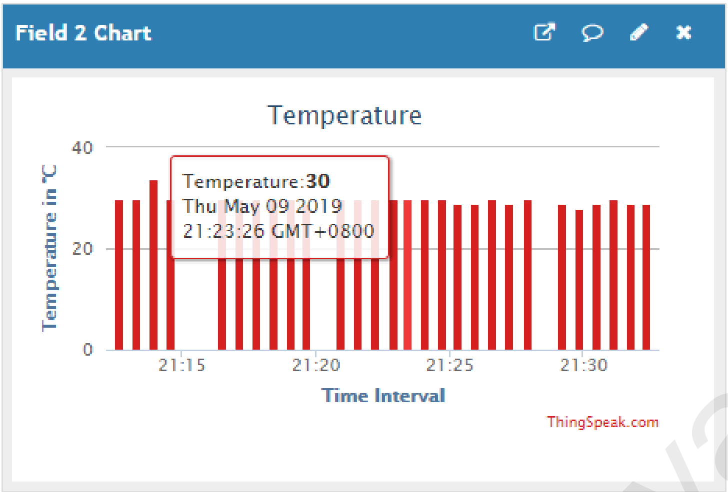


Figure 17: 15th Data

16th data received at 21:24:04

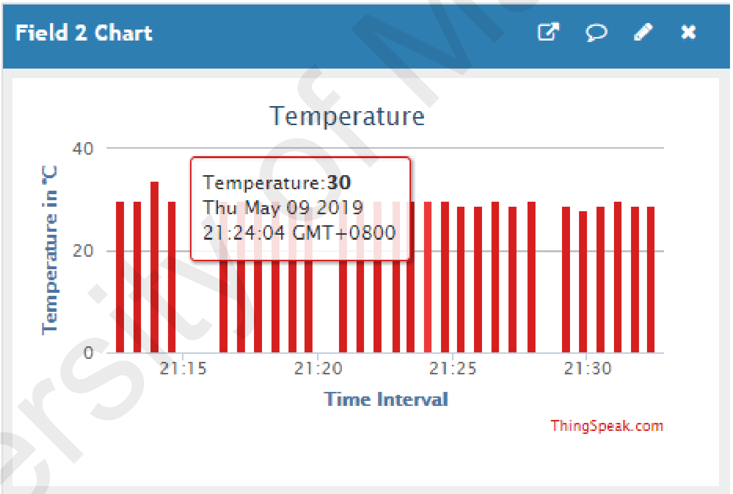


Figure 18: 16th Data

17th data received at 21: 24: 41

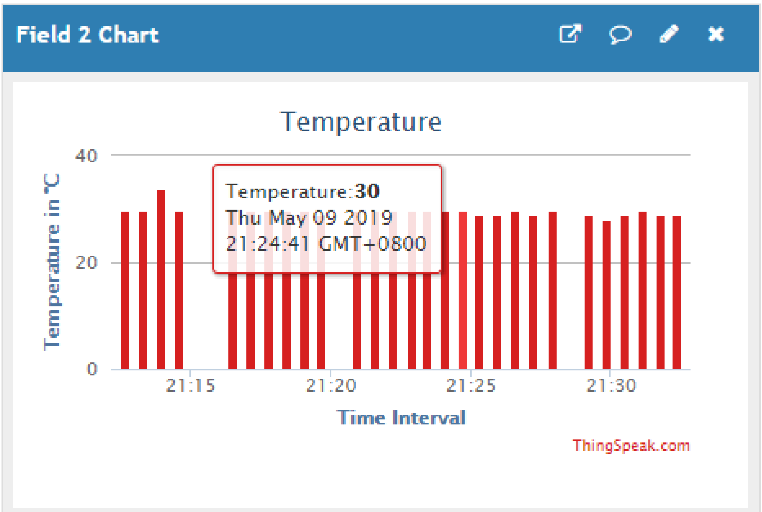


Figure 19: 17th Data

Summary:

Data No.	Received Time	Delay between Each Data (in Seconds)
12th Data	21:21:33	-
13th Data	21:22:11	38
14th Data	21:22:52	41
15th Data	21:23:26	34
16th Data	21:24:04	38
17th Data	21:24:41	37
Average Delay in between each data		(38+41+34+38+37)/5 = 37.6

Table 9: Summary Table of Temperature Sensor at Room

Therefore the average delay for the temperature sensor is 37.6 seconds in this environment.

Time Delay in between both sensors for sending same data:

Data No.	Data Receiving time for Pulse Sensor	Data Receiving time for Temperature Sensor	Delay in between Each Sensor Data (in Seconds)
12th Data	21:20:35	21:21:33	58
13th Data	21:21:14	21:22:11	57
14th Data	21:21:53	21:22:52	59
15th Data	21:22:31	21:23:26	55
16th Data	21:23:09	21:24:04	55
17th Data	21:23:46	21:24:41	55
Average delay in between both sensors’ data			(58+57+59+55+55+55)/6 = 56.5

Table 10: Average delay in between both sensors at room

In this environment, the average delay time in between both sensors is 56.5 seconds.

Test No: 02

Location: Hall Area at Block A, KK12, University of Malaya.

Average Wi-Fi Speed during the Test:

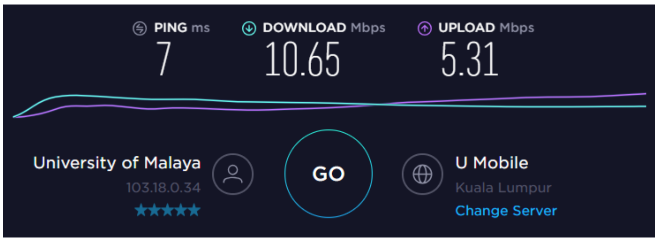


Figure38: Average WI-Fi Speed at the hall

Average Mobile Signal Strength during the Test:

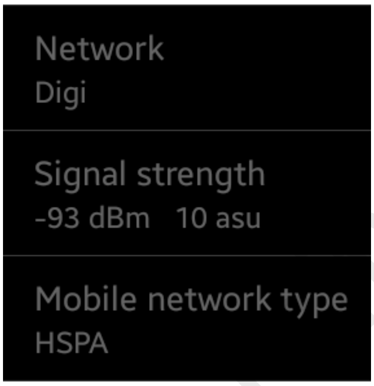


Figure 39: Average Mobile Hotspot Strength at the hall

Data Obtained at ThingSpeak™ IoT Platform:

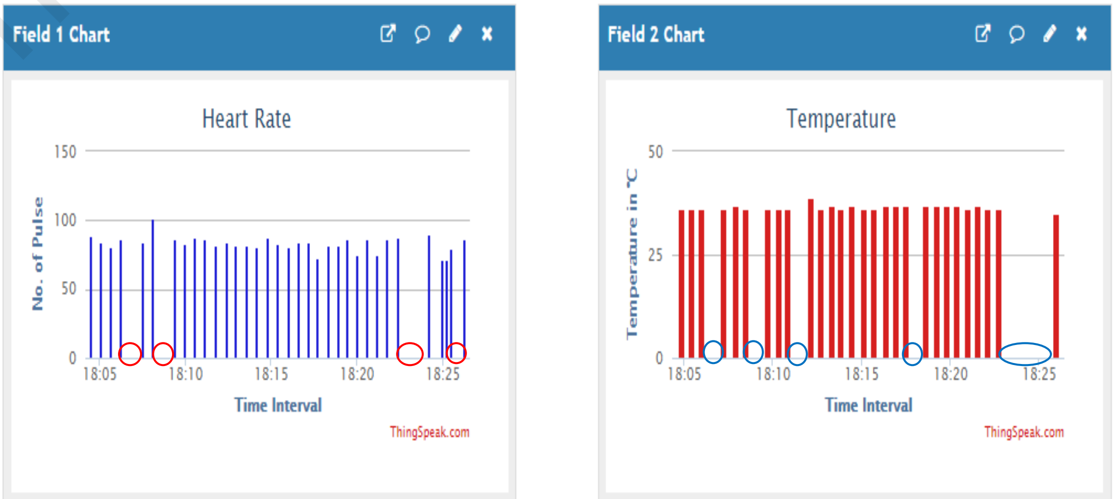


Figure 40: Obtained data at the hall

Observations from the above Data:

Data were recorded from both sensors for same time period E.g. Data for 18.05 - 18.25 can be seen here.

There were some uneven time delays occurred in both Charts which are circled in Red and Blue colors. (In Heart Rate 4 times and in Temperature 5 times).

Values of these sensors are not stable and fluctuate between a certain rages.

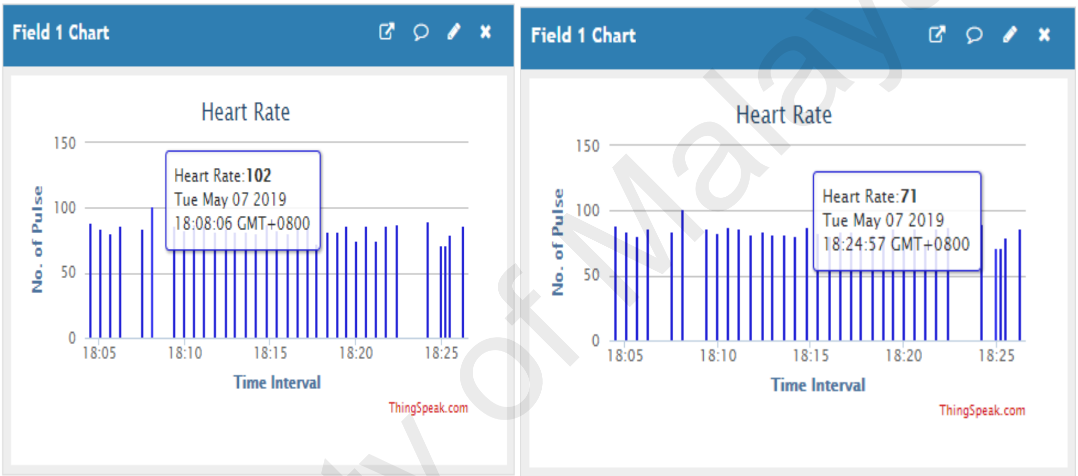


Figure 41: Obtained Pulse sensor data at the hall

Above graphs show the highest and lowest heart rates obtained during the testing time period which are 102 BPM and 71 BPM respectively and all the other values fall in between these values. Likewise below graphs show the highest and the lowest body temperature values obtained during the experimental times which are 39°C and 35°C and the rest of the values fall in between these two values.

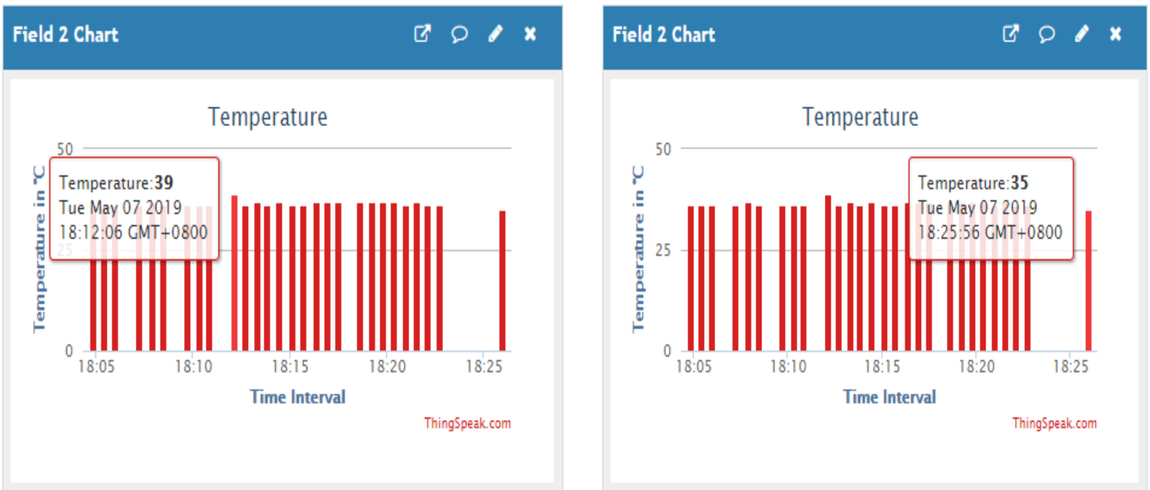


Figure 42: Obtained Temperature Sensor data at the hall

Data Analyzing:

Same above used analyzing technique is applied here. But here starting from data number 10 which was selected randomly and there are no uneven time delays took place after this data in both sensors for some certain extend which provides a number of continuous data for analyzing:

Data for Heart Rate Sensor:

10th data was received at **18:11:08**

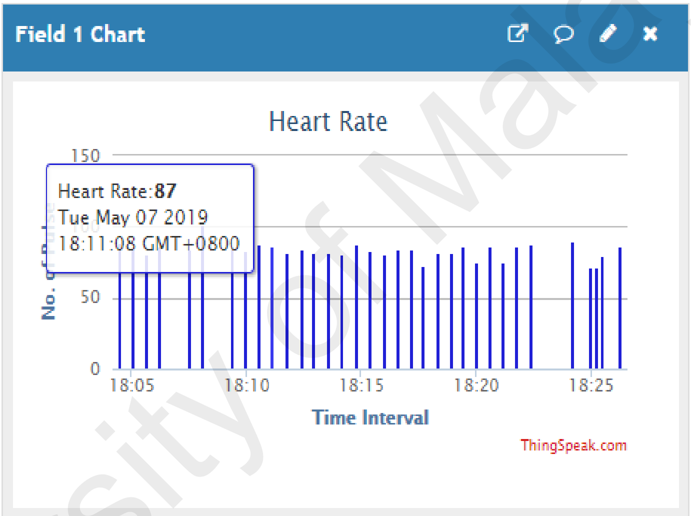


Figure 43: 10th data

11th data was received at **18:11:46**

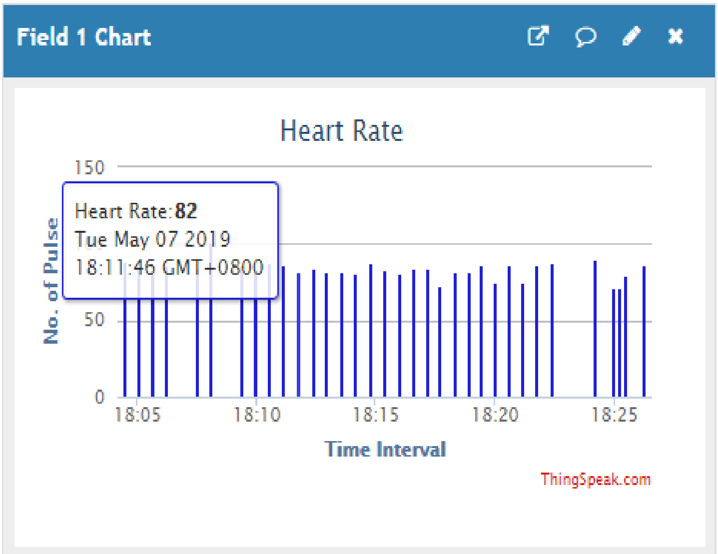


Figure 44: 11th data

12th data was received at **18:12:24**

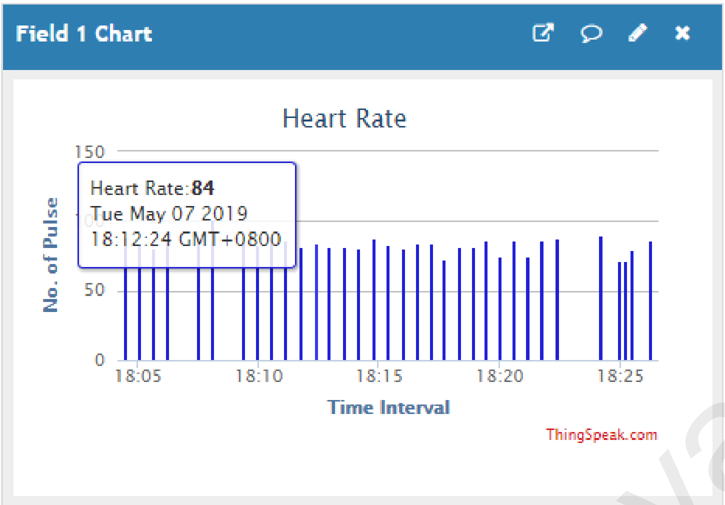


Figure 45: 12th data

13th data was received at **18:12:56**

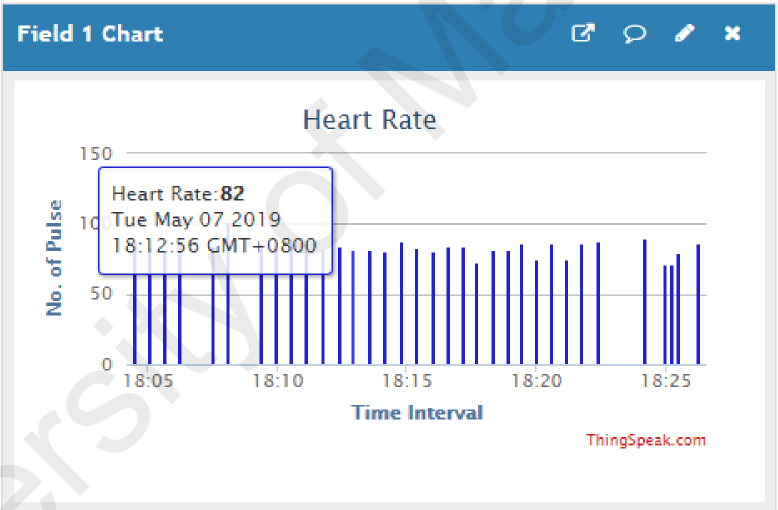


Figure 46: 13th data

14th data was received at **18:13:32**

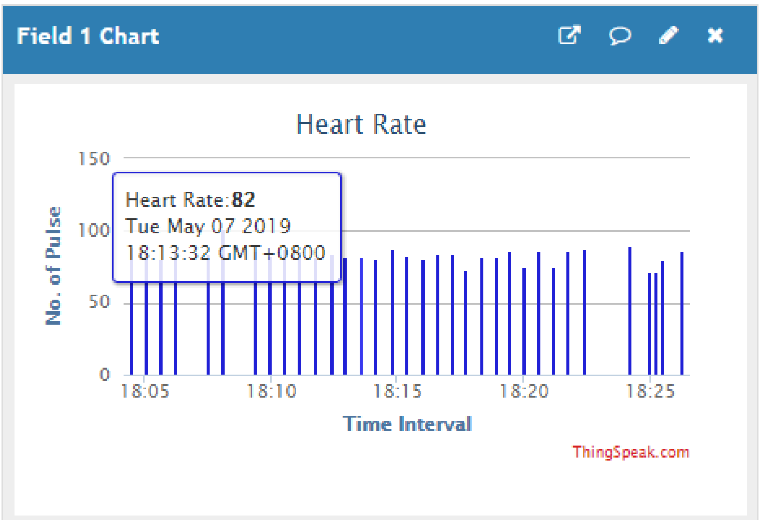


Figure 47: 14th data

15th data was received at **18:14:07**

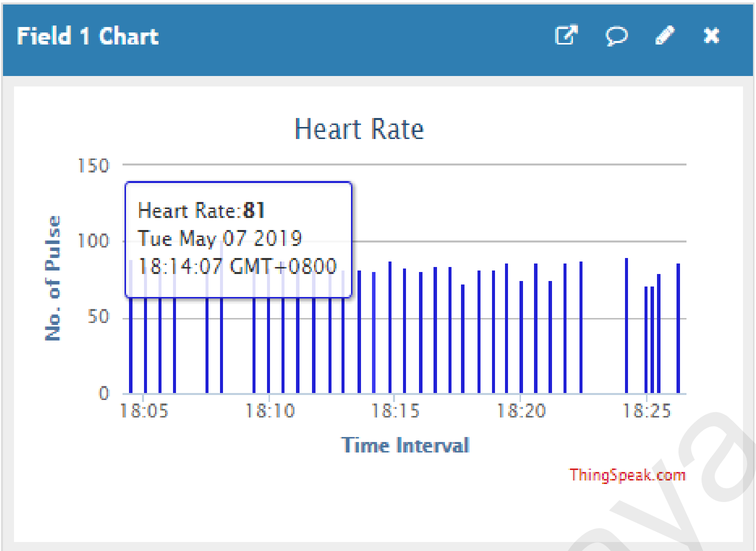


Figure 48: 15th data

Summary:

Data No.	Received Time	Delay between Each Data (in Seconds)
10th Data	18:11:08	-
11th Data	18:11:46	38
12th Data	18:12:24	38
13th Data	18:12:56	32
14th Data	18:13:32	38
15th Data	18:14:07	35
Average Delay in between each data		(38+38+32+38+35)/ 5 = 36.2

Table 11: Summary table of Pulse sensor at the hall

According to this table, the average delay for the pulse sensor data in this environment is 36.2 seconds.

Data for Temperature Senor:-

10th data was received at **18:12:06**

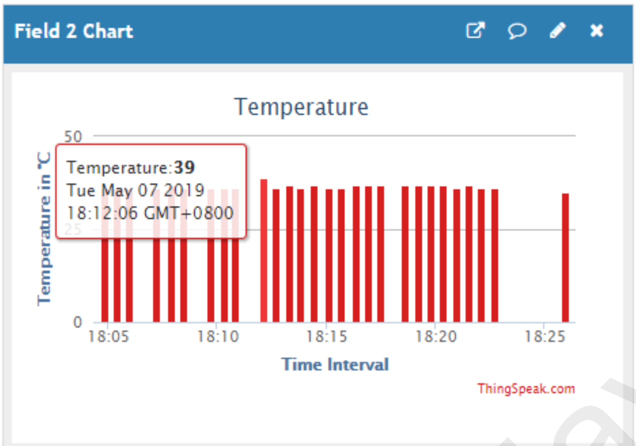


Figure 49: 10 th data

11th data was received at **18:12:40**

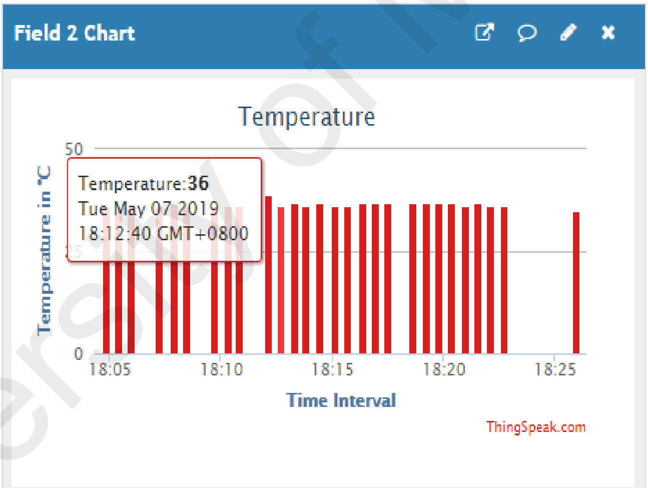


Figure 50: 11th data

12th data was received at **18:13:15**

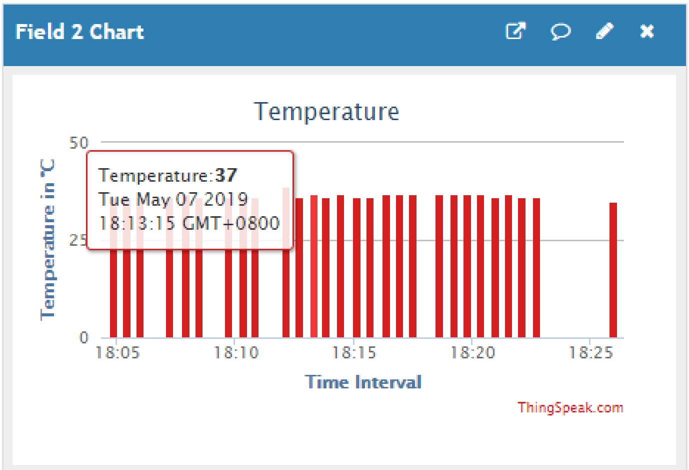


Figure 20: 12th data

13th data was received at **18:13:48**

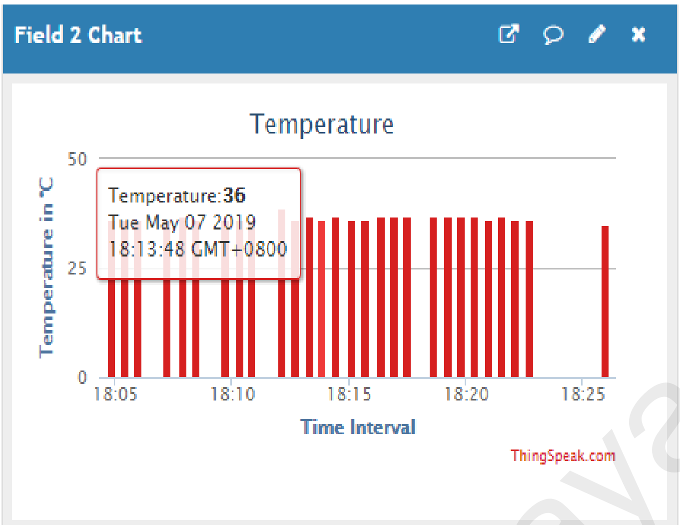


Figure 52: 13th data

14th data was received at **18:14:24**

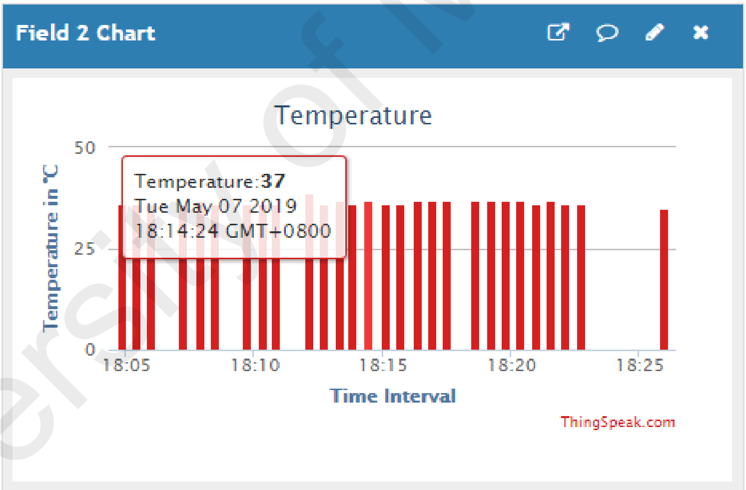


Figure 53: 14th data

15th data was received at **18:15:05**

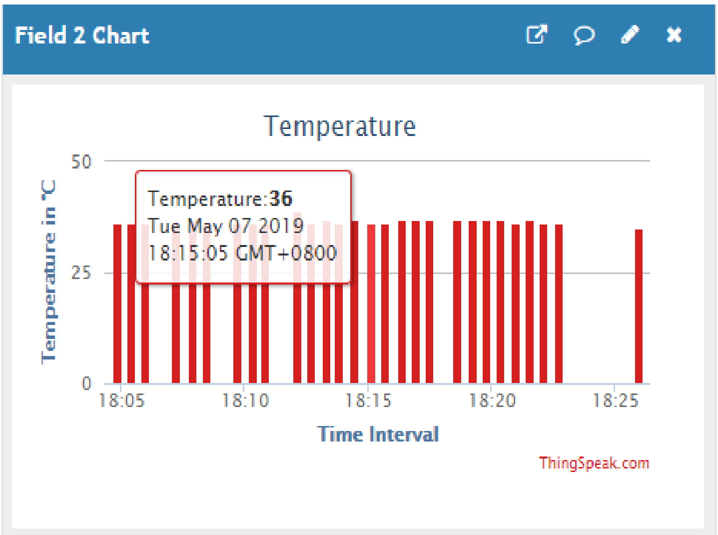


Figure 54: 15th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
10th Data	18:12:06	-
11th Data	18:12:40	34
12th Data	18:13:15	35
13th Data	18:13:48	33
14th Data	18:14:24	36
15th Data	18:15:05	41
Average Delay in between each data		(34+35+33+36+41)/5 = 35.8

Table 12: Summary table of temperature sensors at the hall

Above table shows, that average delay for the temperature sensor at the hall is 35.8 seconds.

Time delay in receiving data from each sensor:

Data No.	Data Receiving time for Pulse Sensor	Data Receiving time for Temperature Sensor	Delay in between Each Sensor Data (in Seconds)
10th Data	18:11:08	18:12:06	58
11th Data	18:11:46	18:12:40	54
12h Data	18:12:24	18:13:15	51
13th Data	18:12:56	18:13:48	52
14th Data	18:13:32	18:14:24	52
15th Data	18:14:07	18:15:05	58
Average delay in between both sensors' data			(58+54+51+52+52+58)/6 = 54.1667

Table 13: Summary table of delay in between both sensors at the hall

As this above table shows that the average delay between both sensor data at this environment is 54.17 seconds.

Test No: 03

Location: Cafeteria, KK12, University of Malaya.

Average Wi-Fi Speed during the Test:

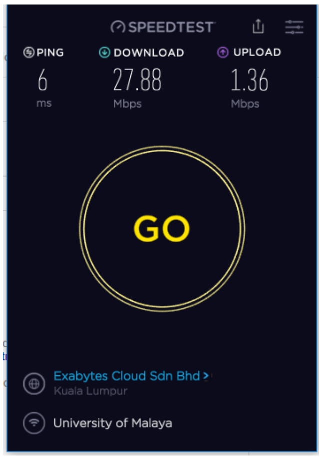


Figure 55: Average Wi-Fi at the Cafeteria

Average Mobile Signal Strength during the Test:

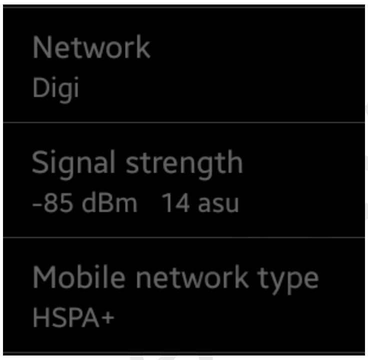


Figure 56: Average Signal Strength at the Cafeteria

Data Obtained at ThingSpeak™ IoT Platform:

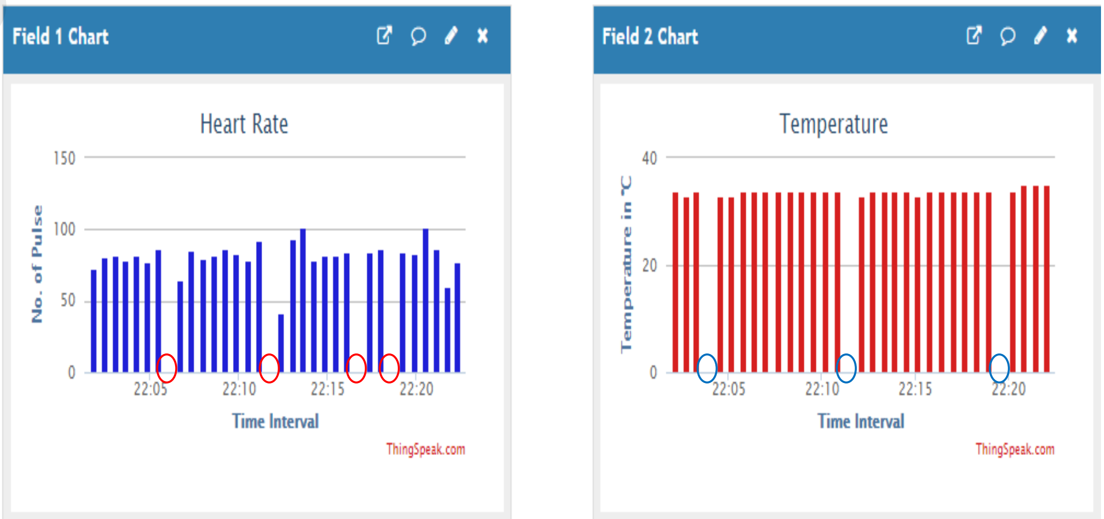


Figure 57: Obtained Data at the Cafeteria

Observations from the above Data:

Data were collected in between same time interval from both sensors. E.g. Data for 22.05 – 22.20 can be obtained here.

Some uneven time delays occurred in both Charts which are circled in Red and Blue colors. (In Heart Rate 4 times and in Temperature 3 times)

Values of these sensors were not stable and fluctuate between a certain rages. The lowest and the height amount received for heart beat and temperature are 42BPM – 102BPM and 33℃ - 35℃. The rest of the values obtained in this graph fall in between these number ranges.

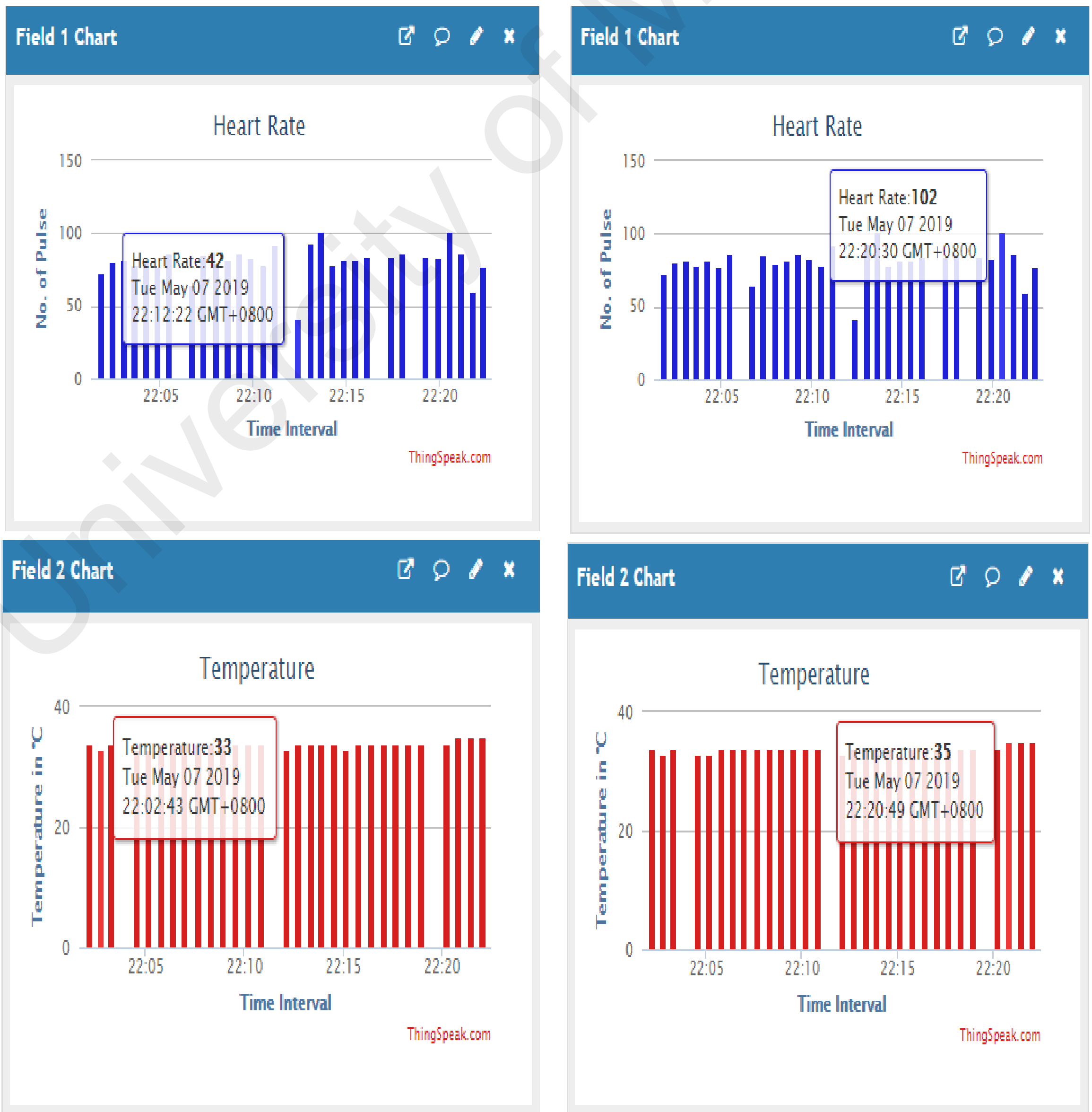


Figure 58: Obtained Heart Rate and Temperature Data at the Cafeteria

Data Analyzing:

Same method is used here. Here data number 8 is selected randomly.

Heart Rate Data:

8th data received at **22:06:39**

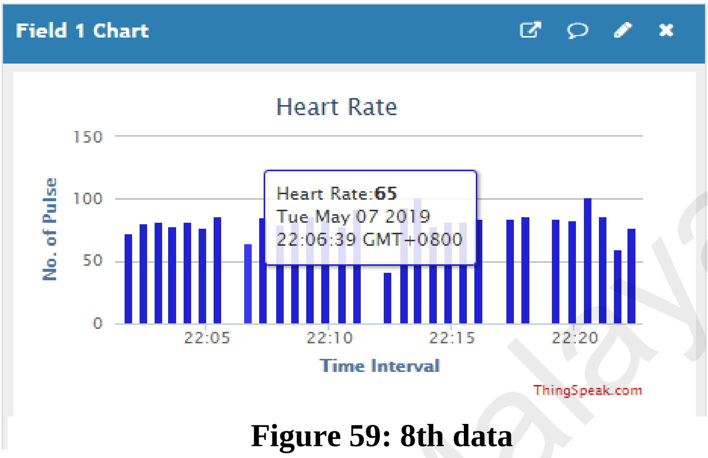


Figure 59: 8th data

9th data received at **22:07:17**

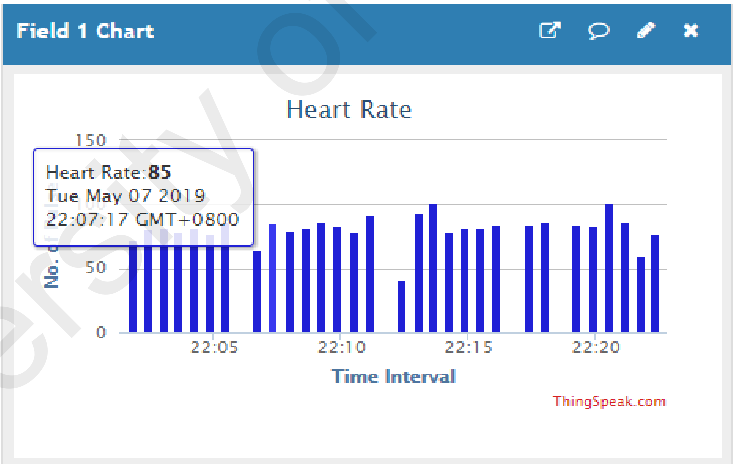


Figure 60: 9th data

10th data received at **22:07:58**

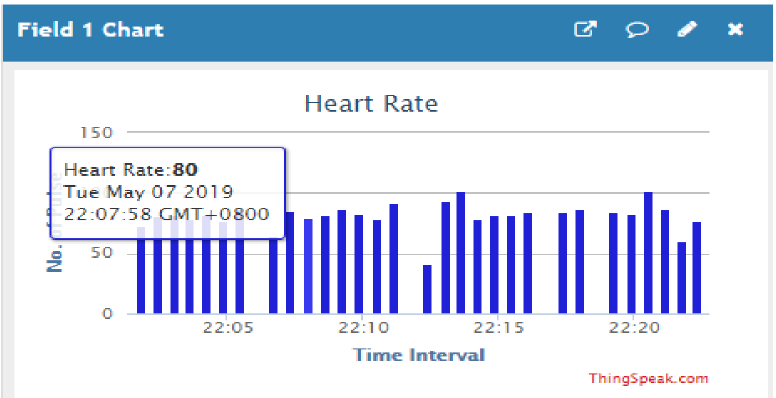


Figure 61: 10th data

11th data received at **22:08:33**

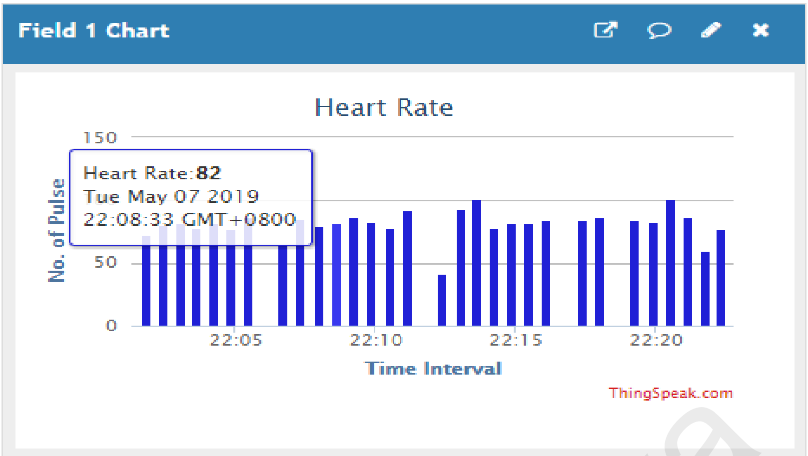


Figure 62: 11th data

12th data received at **22:09:12**

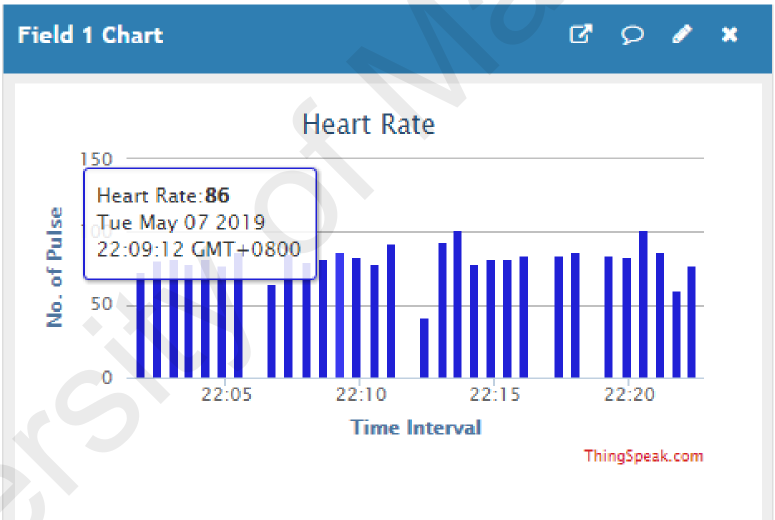


Figure 63: 12th data

13th data received at **22:09:48**

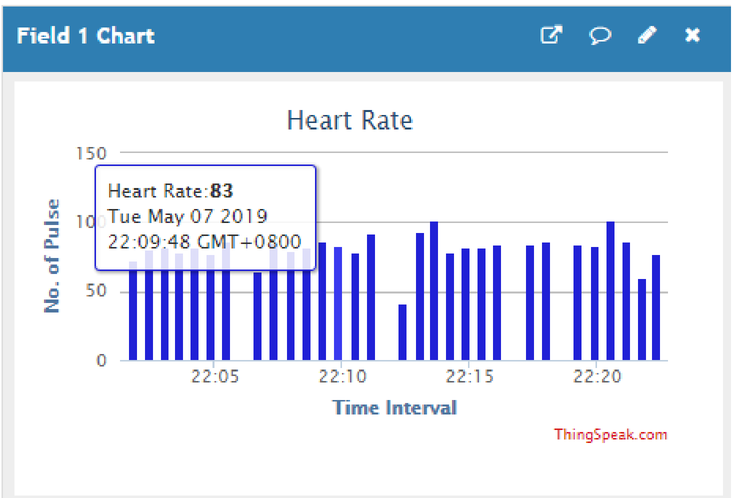


Figure 64: 13th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
8th Data	22:06:39	-
9th Data	22:07:17	38
10th Data	22:07:58	41
11th Data	22:08:33	35
12th Data	22:09:12	39
13th Data	22:09:48	36
Average Delay in between each data		(38+41+35+39+36)/5 = 37.8

Table 14: Summary table of Pulse sensor data at the Cafeteria

As this table shows, the average delay for the pulse sensor data is 37.8 seconds.

Temperature Data:

8th data received at **22:06:57**

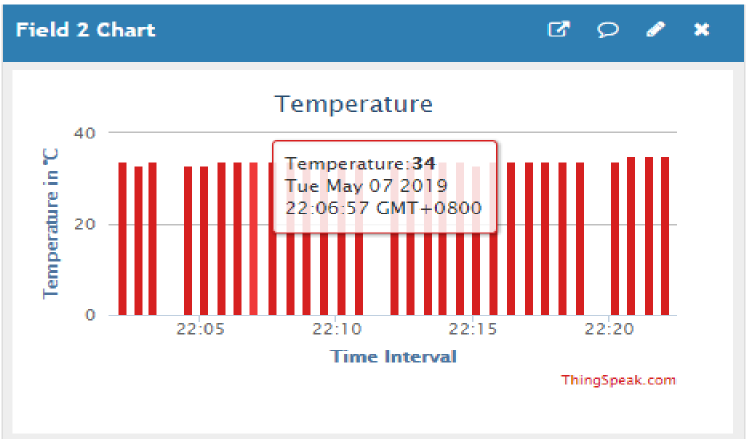


Figure 65: 8th data

9th data received at **22:07:37**

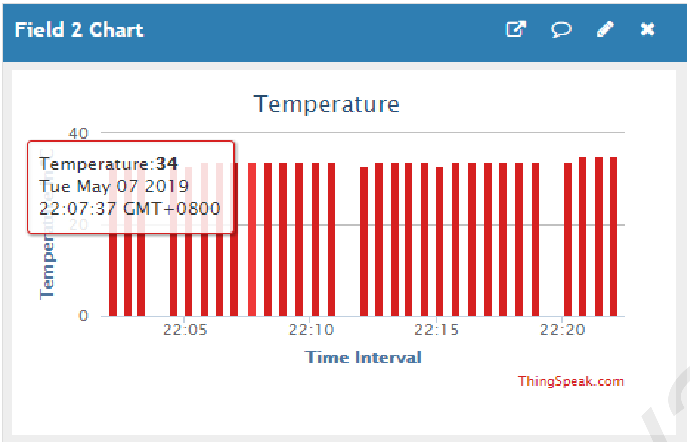


Figure 66: 9th data

10th data received at **22:08:17**

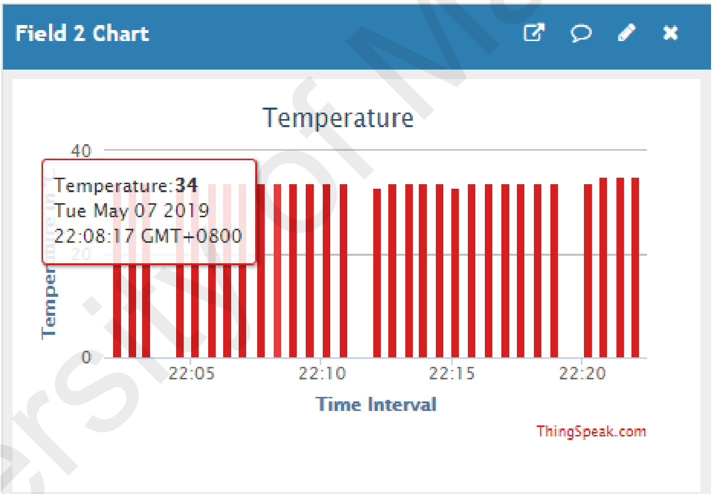


Figure 67: 10th data

11th data received at **22:08:52**

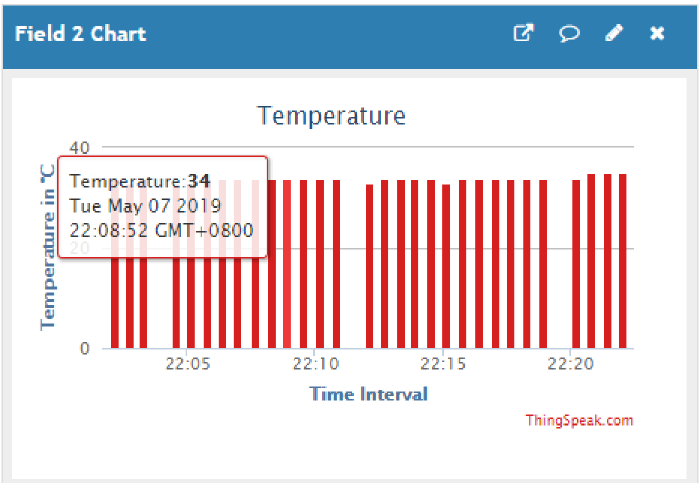


Figure 68: 11th data

12th data received at **22:09:30**

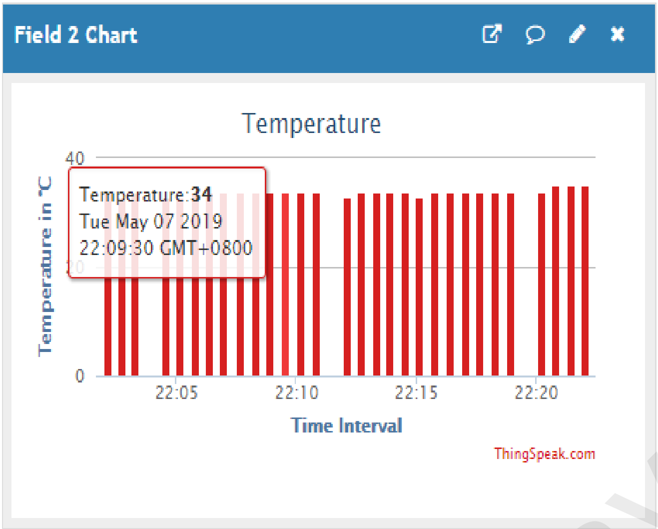


Figure 69: 12th data

13th data received at **22:10:09**

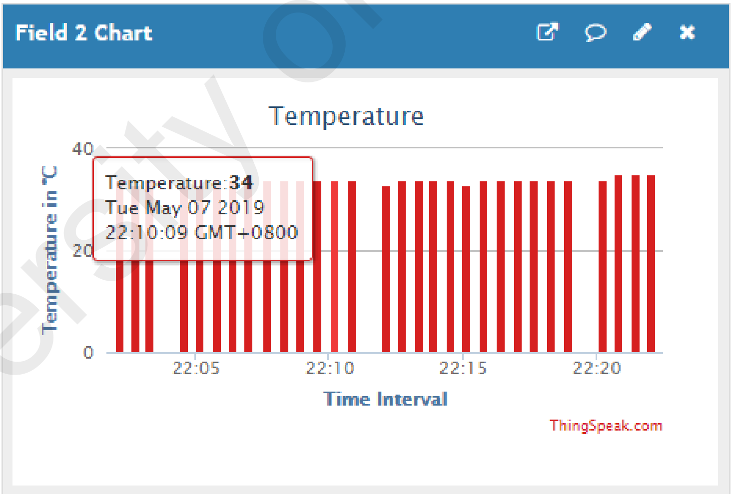


Figure 70: 13th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
8th Data	22:06:57	-
9th Data	22:07:37	40
10th Data	22:08:17	40
11th Data	22:08:52	35
12th Data	22:09:30	38
13th Data	22:10:09	39
Average Delay in between each data		(40+40+35+38+39)/5 = 38.4

Table 4: Summary table of Temperature Sensor data at the Cafeteria

As this table shows the average time delay between each data of temperature sensor at the cafeteria is 38.4 seconds.

Time Delay in between both sensors for sending same data:

Data No.	Data Receiving time for Pulse Sensor	Data Receiving time for Temperature Sensor	Time Delay in between Each Sensor Data (in Seconds)
8th Data	22:06:39	22:06:57	17
9th Data	22:07:17	22:07:37	20
10th Data	22:07:58	22:08:17	19
11th Data	22:08:33	22:08:52	19
12th Data	22:09:12	22:09:30	18
13th Data	22:09:48	22:10:09	21
Average delay in between both sensors' data			(17+20+19+19+18+21)/6 = 19

Table 5: Summary table of delay in between both sensors at the Cafeteria

As the above table shows that the average delays in between both sensors are 19 seconds at the Cafeteria.

Test No: 04

Location: University of Malaya Main Library – 4th Floor.

Average Wi-Fi Speed during the Test:

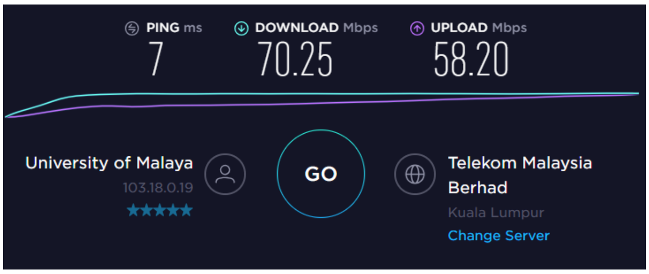


Figure 71: Average wi-fi speed at the library

Average Mobile Signal Strength during the Test:

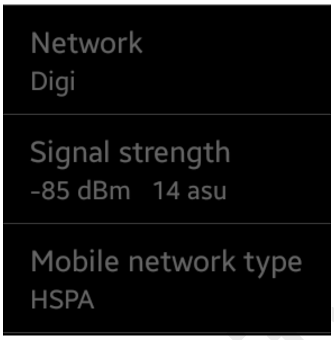


Figure72: Average Signal Strength at the library

Data Obtained at ThingSpeak™ IoT Platform:

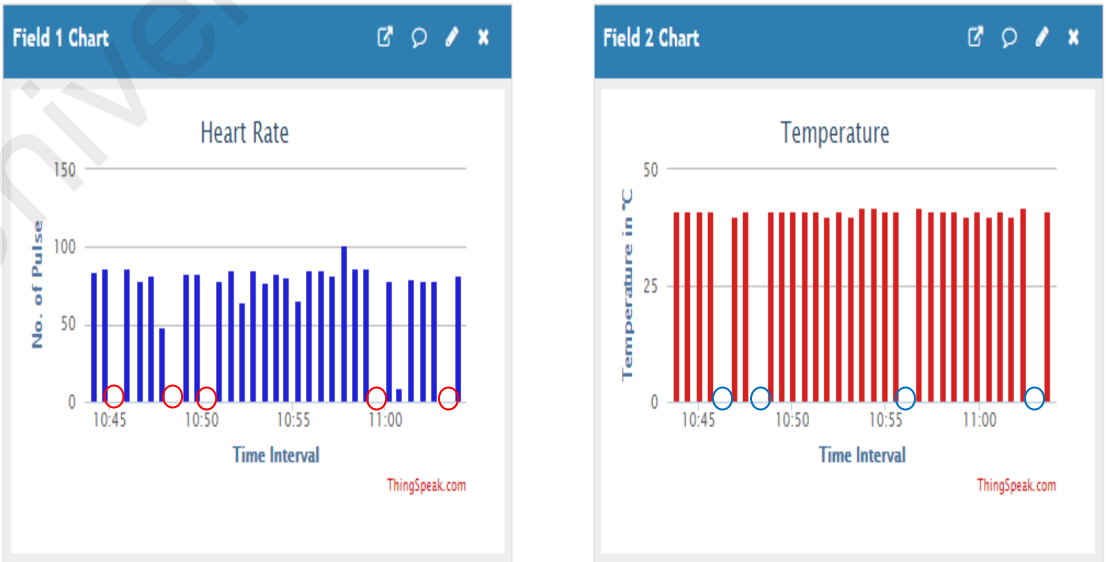


Figure 73: Obtained data at the Library

Data for same time period were collected. E.g. Data between 10:45 – 11:00 can be seen here.

Some uneven time delays occurred in both Charts which are circled in Red and Blue colors. (In Heart Rate 5 times and in Temperature 4 times)

Values of these sensors are not stable and deviate from 49 BPM – 102 BPM for Heart Rate and the rest of the values fall in between these two limits. Likewise Temperature varies 40℃ - 42℃ and the rest of the readings fall in between these numbers.

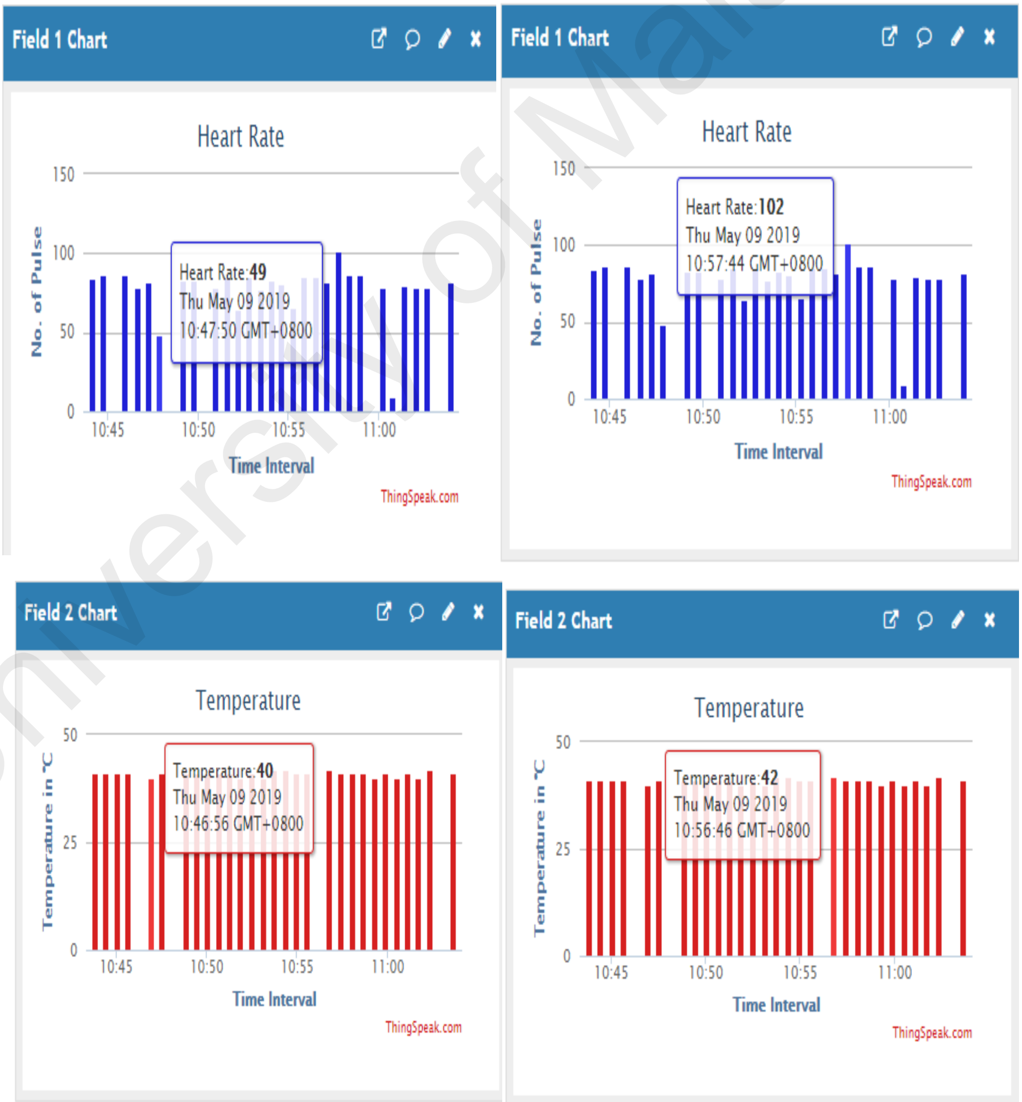


Figure 74: Pulse Rate and Temperature Sensors data

Same analyzing technique is used here. Here data number 11 is selected randomly.

Heart Rate Data:

11th Data received at **10:52:10**

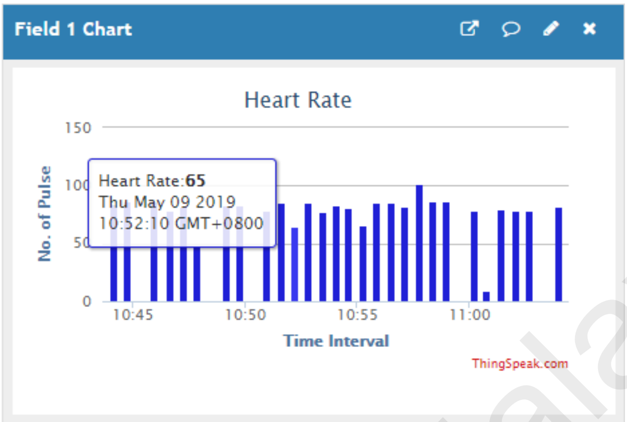


Figure 75: 11th data

12th Data received at **10:52:48**

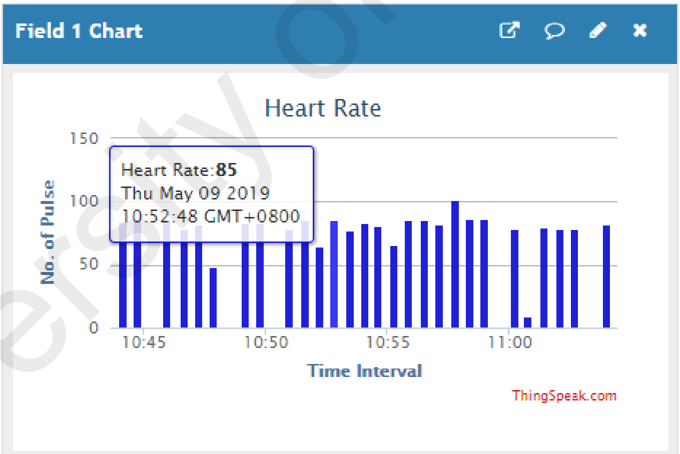


Figure 76: 12th data

13th Data received at **10:53:26**

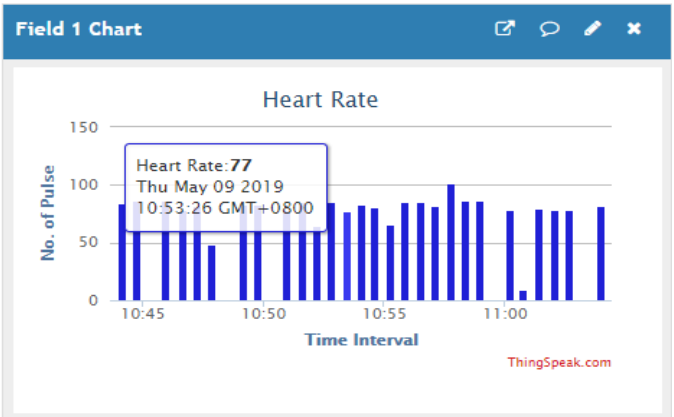


Figure 77: 13th data

14th Data received at **10:54:01**

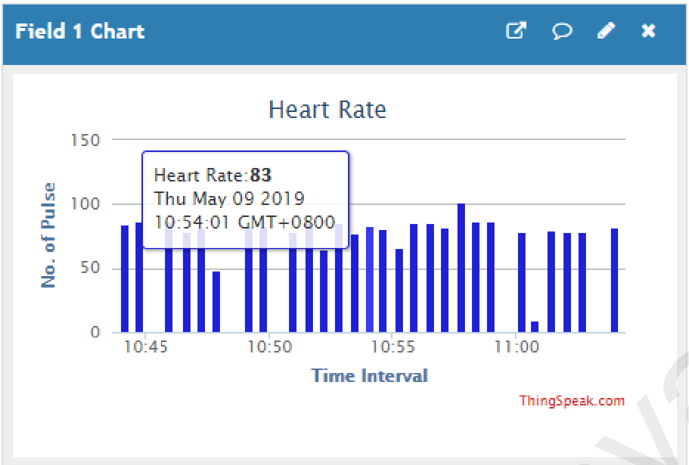


Figure 78: 14th data

15th Data received at **10:54:36**

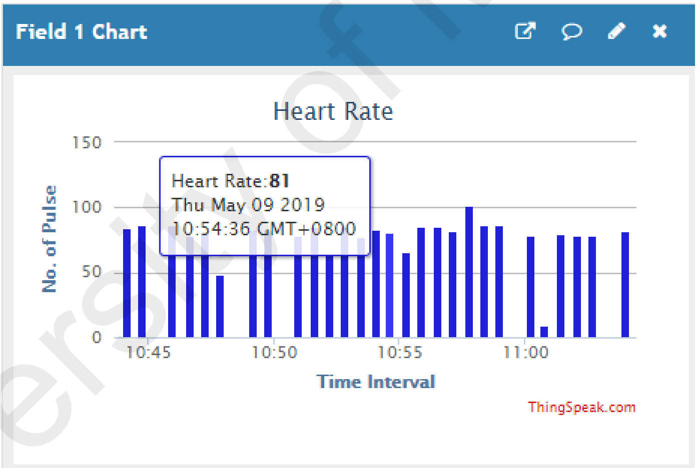


Figure 79: 15th data

16th Data received at **10:55:14**

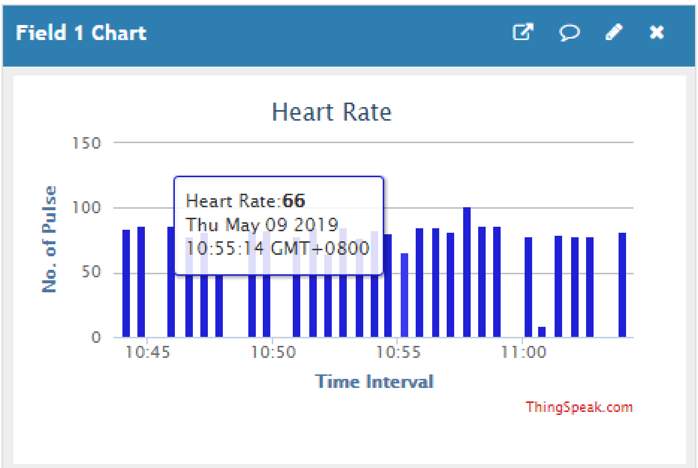


Figure 80: 16th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
11th Data	10:52:10	-
12^{h} Data	10:52:48	38
13^{h} Data	10:53:26	38
14th Data	10:54:01	35
15th Data	10:54:36	35
16th Data	10:55:14	38
Average Delay in between each data		(38+38+35+35+38)/5 = 36.8

Table 6: Summary table of Pulse sensor at the library

As shown above, the average time delay between each data of pulse sensor at the library is 36.8 seconds.

Temperature Data:

11th Data received at 10:51:15

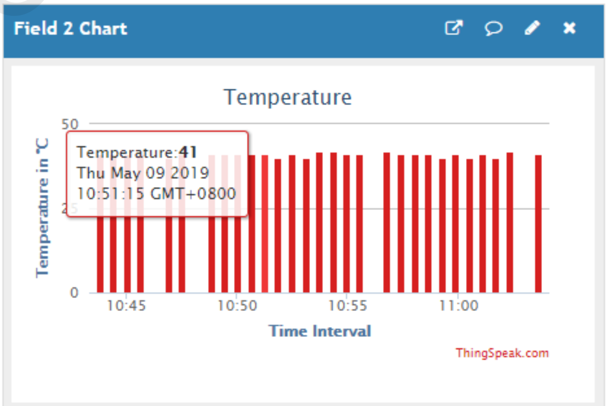


Figure 81: 11th data

12th Data received at 10:51:51

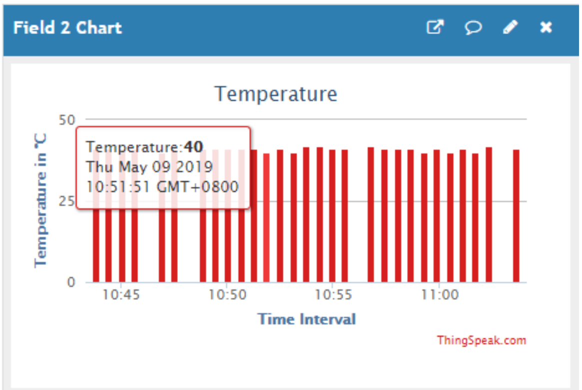


Figure 82: 12th data

13th Data received at 10:52:29

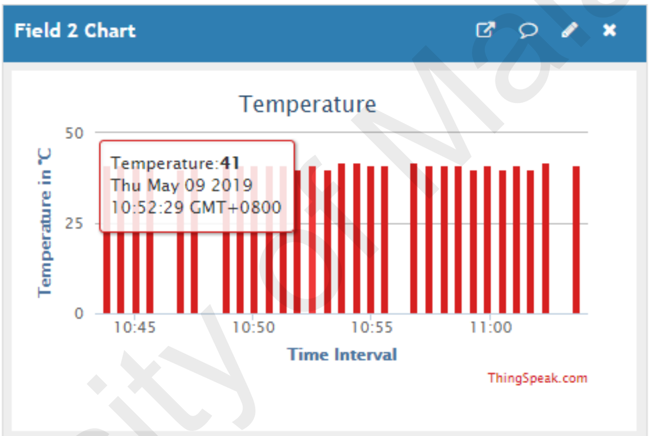


Figure 83: 13th data

14th Data received at 10:53:06

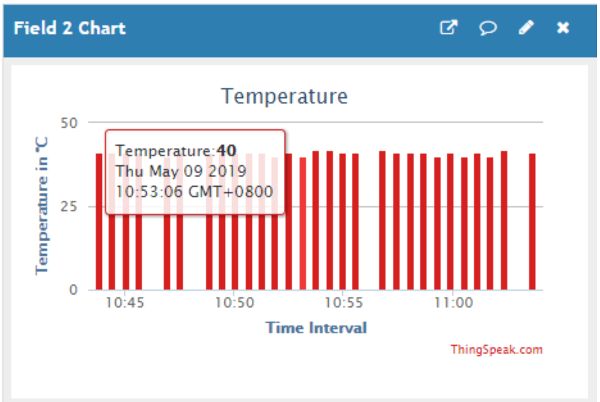


Figure 84: 14th data

15th Data received at 10:53:44

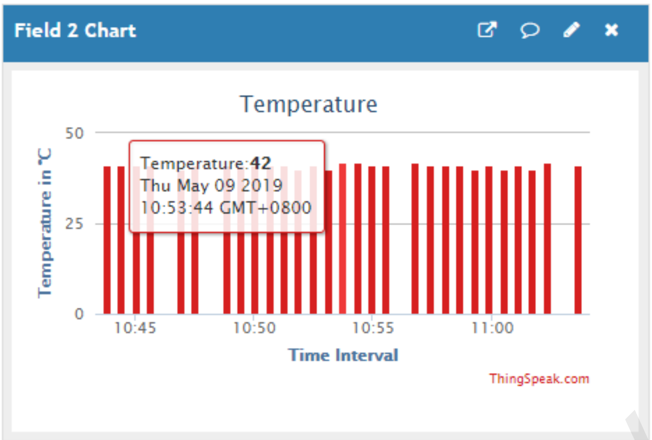


Figure 85: 15th data

16th Data received at 10:54:20

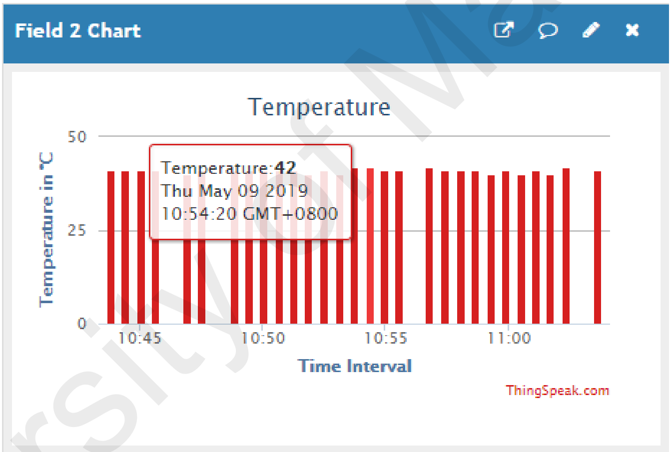


Figure 86: 16th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
11th Data	10:51:15	-
12h Data	10:51:51	36
13h Data	10:52:29	38
14th Data	10:53:06	37
15th Data	10:53:44	38

16th Data	10:54:20	36
Average Delay in between each data		(36+38+37+38+36)/5 = 37

Table 7: Summary table of average delay in temperature sensor data at the library

This table shows that the average delay in between each data of the temperature sensor is 37 seconds.

Time Delay in between both sensors for sending same data:

Note that in this test data from temperature were received before pulse sensor.

Data No.	**Data Receiving time for Temperature Sensor**	**Data Receiving time for Pulse Sensor**	**Time Delay in between Each Sensor Data (in Seconds)**
11th Data	10:51:15	10:52:10	55
12h Data	10:51:51	10:52:48	57
13h Data	10:52:29	10:53:26	57
14th Data	10:53:06	10:54:01	55
15th Data	10:53:44	10:54:36	52
16th Data	10:54:20	10:55:14	54
Average delay in between both sensors' data			(55+57+57+55+52+54)/6 = 55

Table 8: Summary table of average time delay in between both sensor data at the library

According to this table, average time delay in between the both sensors to send the data is 55 seconds.

Test No: 05

Location: Study Hall in Engineering Tower at University of Malaya.

Average Wi-Fi Speed during the Test:

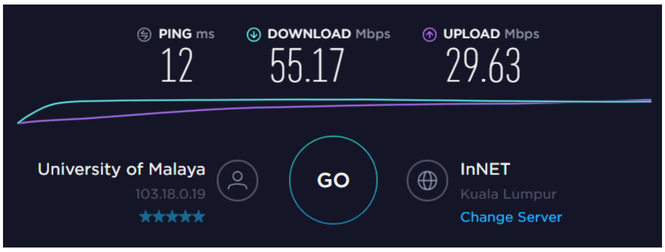


Figure 87: Average wi-fi speed at the

Mobile Signal Strength during the Test:

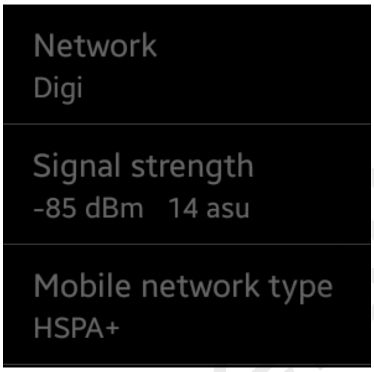


Figure 88: Average Mobile signal Strength at the engineering tower

Data Obtained at ThingSpeak™ IoT Platform:

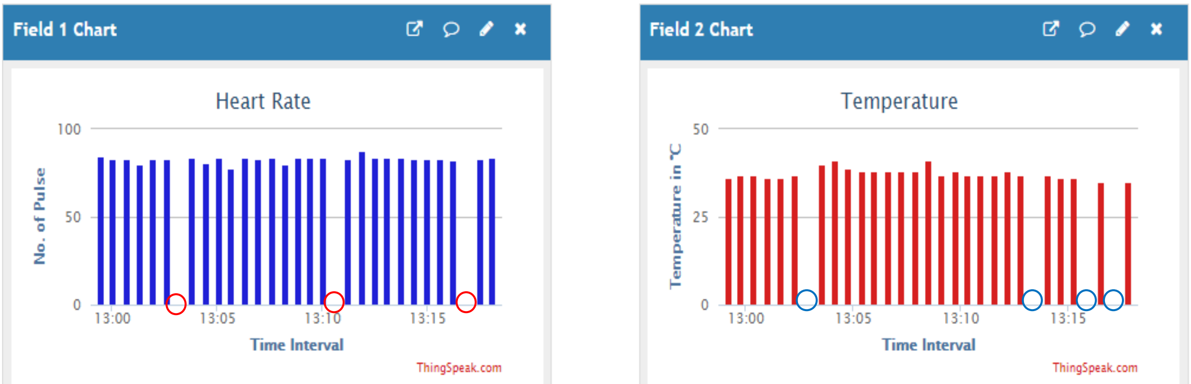


Figure 89: Obtained data at the engineering tower

Observations from the above Data:

Data from both sensors for certain time period were obtained. E.g. In this case Data were observed between 13:00 – 13:15.

There were some uneven time delays occurred in both Charts which are circled in Red and Blue colors. (In Heart Rate 3 times and in Temperature 4 times)

Values of these sensors are not constant. Heart Rate varies from 78 BPM – 88 BPM and the rest of the values fall in between these two limits. Likewise Temperature fluctuates from 36°C - 41°C and the rest of the readings fall in between these numbers.

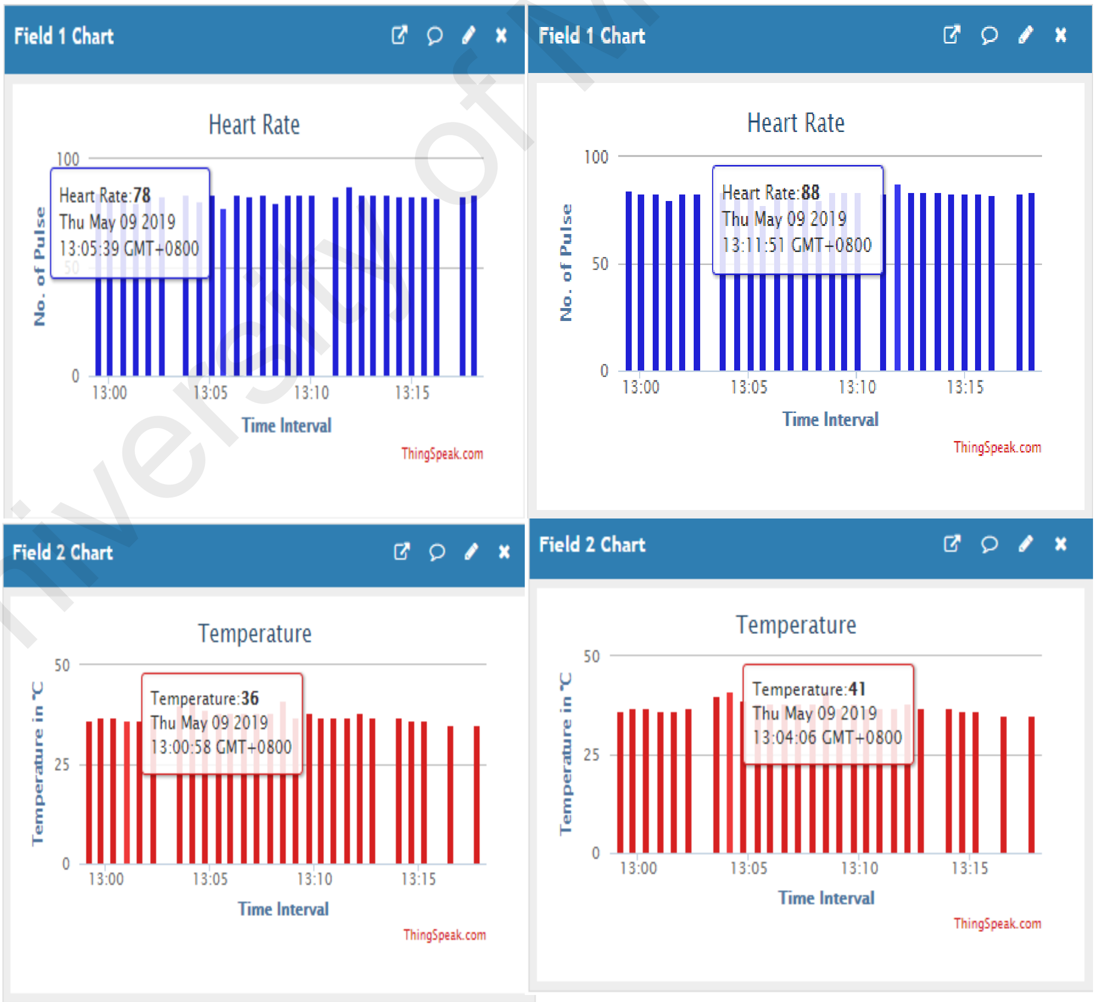


Figure 21: Pulse sensor and temperature sensor data at the engineering tower

Data Analyzing:

Same method is used here. Randomly data number 8 is selected here.

Heart Rate Data:

8th data received at 13:04:26

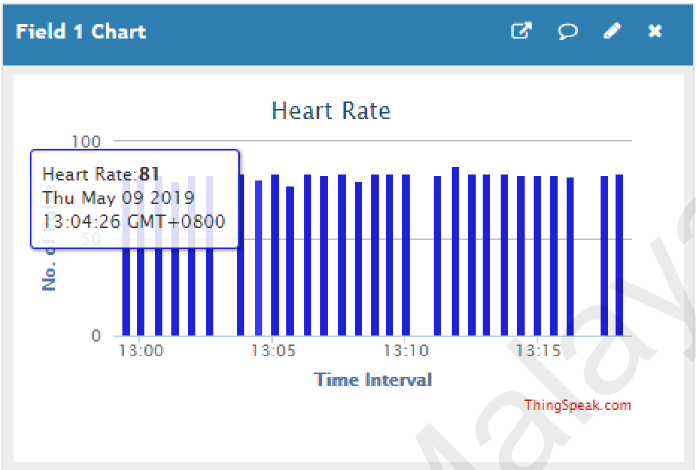


Figure 90: 8th data

9th data received at 13:05:04

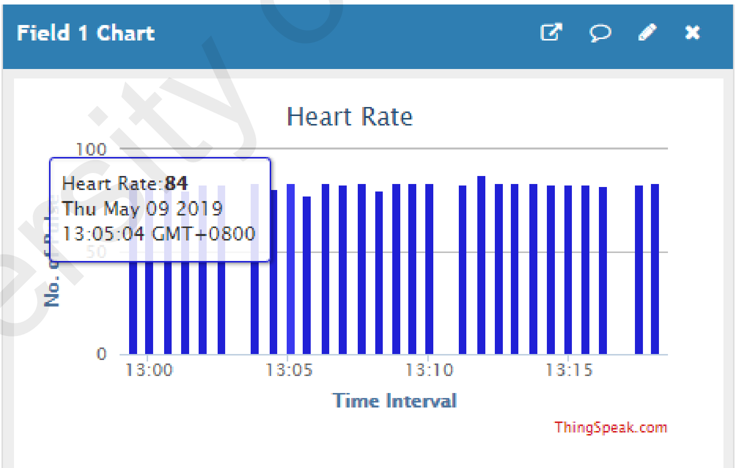


Figure 91: 9th data

10th data received at 13:05:39

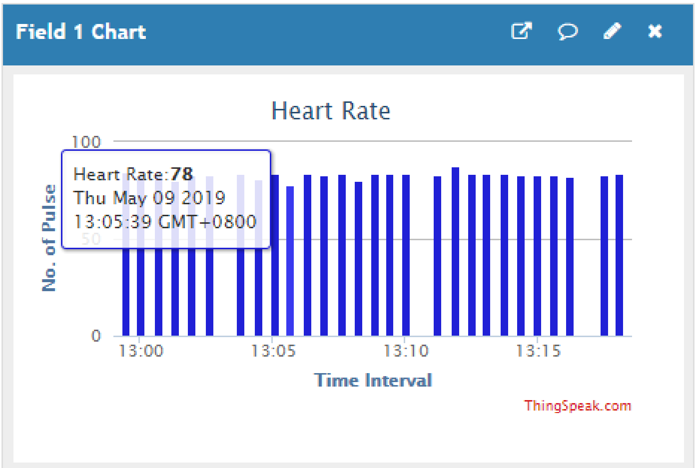


Figure 92: 10th data

11th data received at 13:06:17

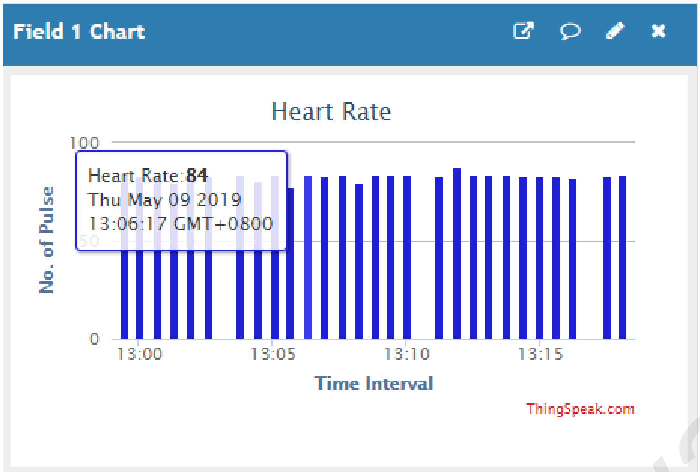


Figure 93: 11th data

12h data received at 13:06:55

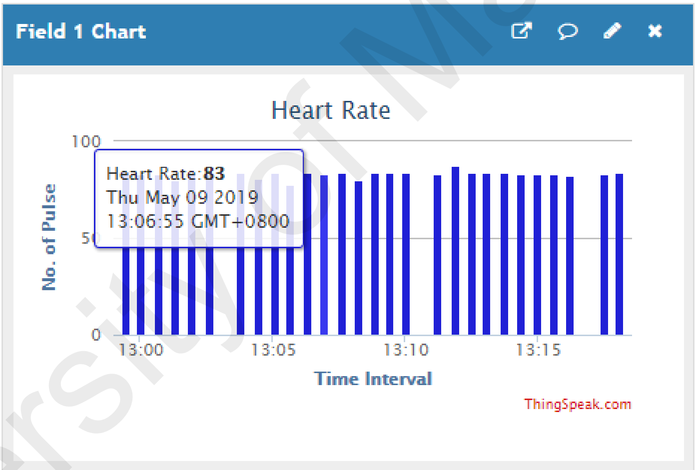


Figure 94: 12th data

13th data received at 13:07:34

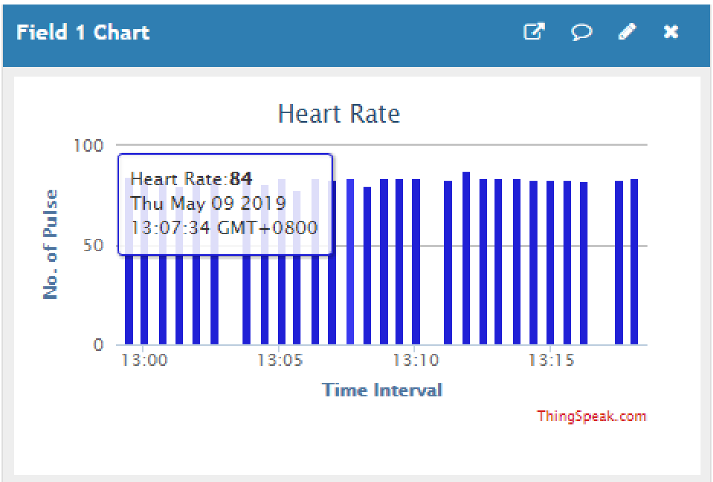


Figure 95: 13th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
8th Data	13:04:26	-
9th Data	13:05:04	38
10th Data	13:05:39	35
11th Data	13:06:17	38
12th Data	13:06:55	38
13th Data	13:07:34	39
Average Delay in between each data		(38+35+38+38+39)/5 = 37.6

Table 9 : Summary table of average data delay for the pulse sensor

As the table shows, that the average delay for the pulse sensor at the engineering tower is 37.6 seconds.

Temperature Data:

8th data received at 13:04:06

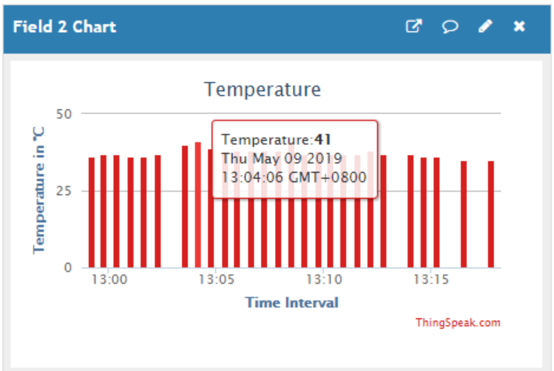


Figure 96: 8th data

9th data received at 13:04:45

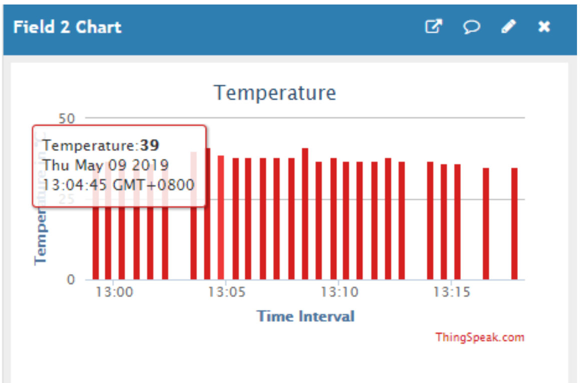


Figure 97: 9th data

10th data received at 13:05:23

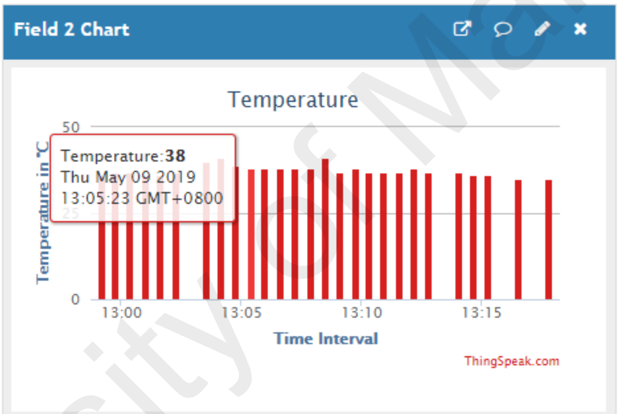


Figure 98: 10th data

11th data received at 13:05:58

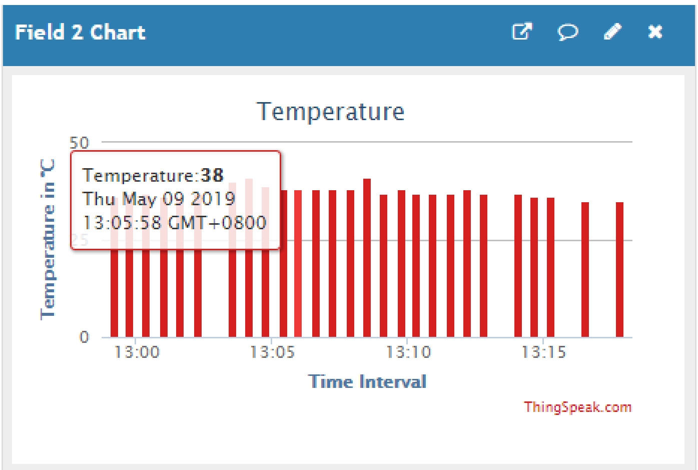


Figure 99: 11th data

12th data received at 13:06:36

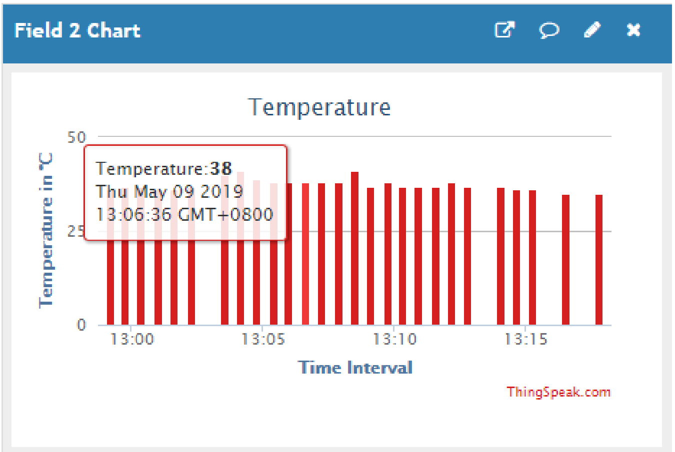


Figure 22: 12th data

13th data received at 13:07:14

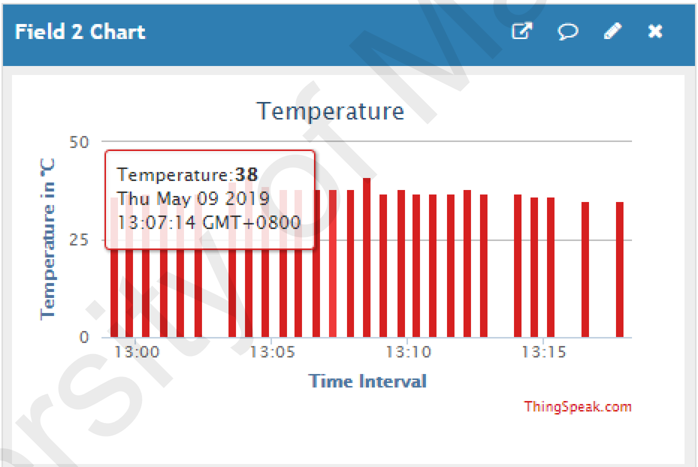


Figure 23: 13th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
8th Data	13:04:06	-
9th Data	13:04:45	39
10th Data	13:05:23	38
11th Data	13:05:58	35
12th Data	13:06:36	38

13th Data	13:07:14	38
Average Delay in between each data		(39+38+35+38+38)/5 = 37.6

Table 10: Summary table of average delay of temperature sensor

Above table shows the average delay in between each data of temperature sensor at the engineering tower is 37.6 seconds.

Time Delay in between both sensors for sending same data:

In this case also Temperature sensor sent the signals before Pulse sensor.

Data No.	Data Receiving time for Temperature Sensor	Data Receiving time for Pulse Sensor	Time Delay in between Each Sensor Data (in Seconds)
8th Data	13:04:06	13:04:26	20
9th Data	13:04:45	13:05:04	19
10th Data	13:05:23	13:05:39	16
11th Data	13:05:58	13:06:17	19
12th Data	13:06:36	13:06:55	19
13th Data	13:07:14	13:07:34	20
Average delay in between both sensors' data			(20+19+16+19+19+20)/6 = 18.8333

Table 11: Summary table of average delay in between both sensors

As the above table shows, the average time delay in between both sensor data is 18.83 seconds at the engineering tower.

Test No: 06

Location: Common Area at IPS, University of Malaya.

Average Wi-Fi Speed during the Test:

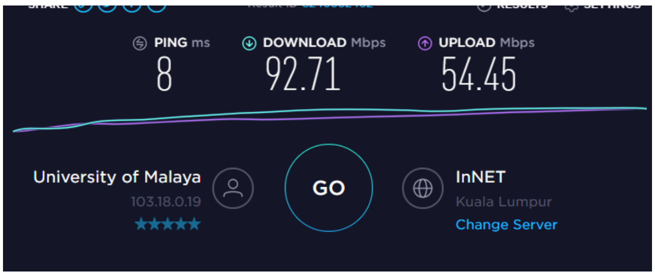


Figure 24: Average wi-fi speed at IPS

Average Mobile Signal Strength during the Test:

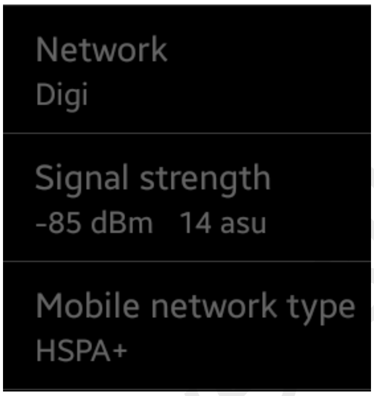


Figure 25: Average mobile signal strength at IPS

Data Obtained at ThingSpeak™ IoT Platform:

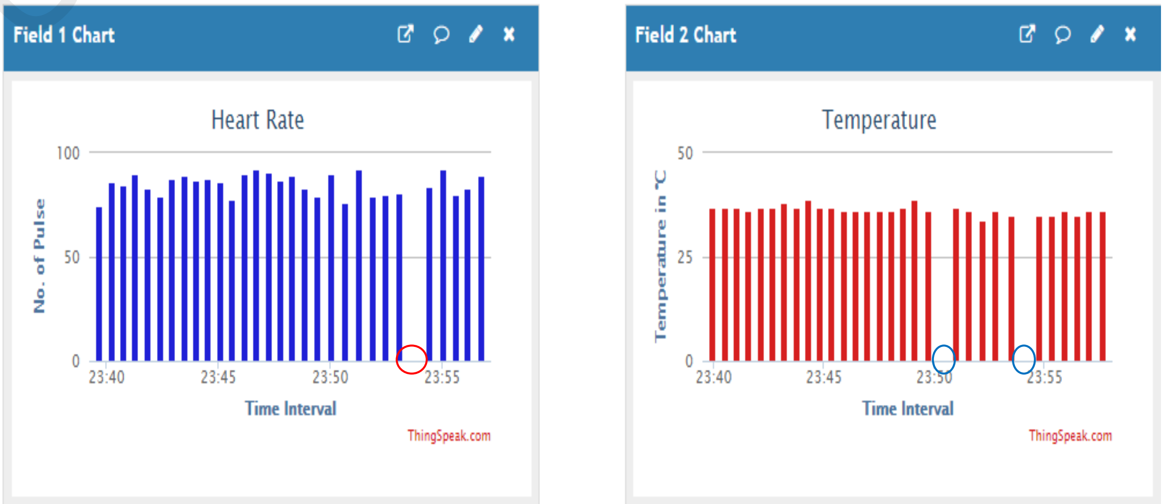


Figure 26: Obtained data at IPS

Observations from the above Data:

Data obtained for same time gap from both sensors. E.g. Data between 23:40 – 23:55 can be observed here.

Some uneven time delays occurred in both Charts which are circled in Red and Blue colors. (In Heart Rate 1 times and in Temperature 2 times)

Sensor values are not stable and fluctuate in between a range. Heart Rate varies from 75 BPM – 92 BPM during the focused time frame and the rest of the values fall in between these two limits. Likewise Temperature 34°C - 39°C and the rest of the readings fall in between these numbers.

Figure 27: Heart Rate and Temperature sensor data

Data Analyzing:

Same method is used here. Randomly data number 3 is selected here.

Heart Rate Data:

3rd data received at 23:40:45

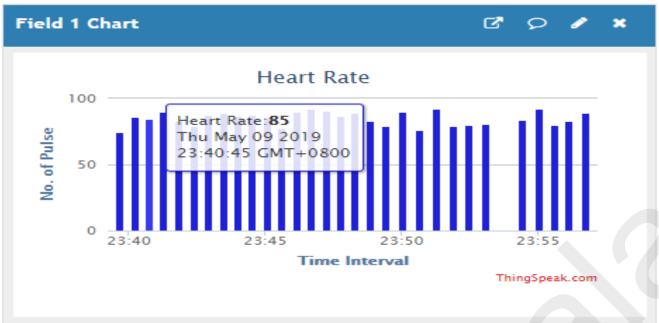


Figure 28: 3rd data

4th data received at 23:41:17

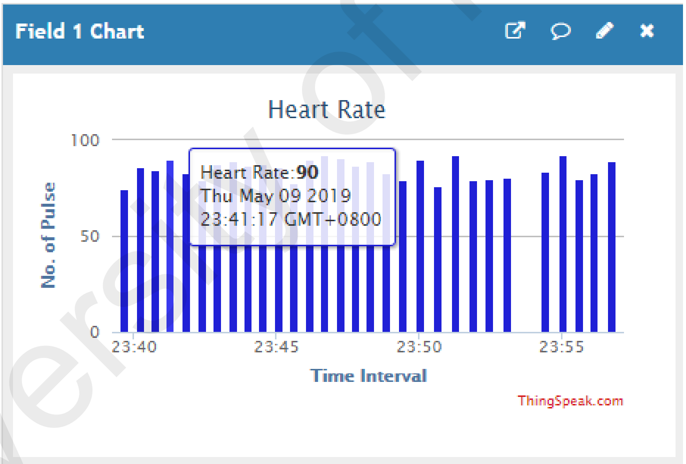


Figure 29: 4th data

5th data received at 23:41:50

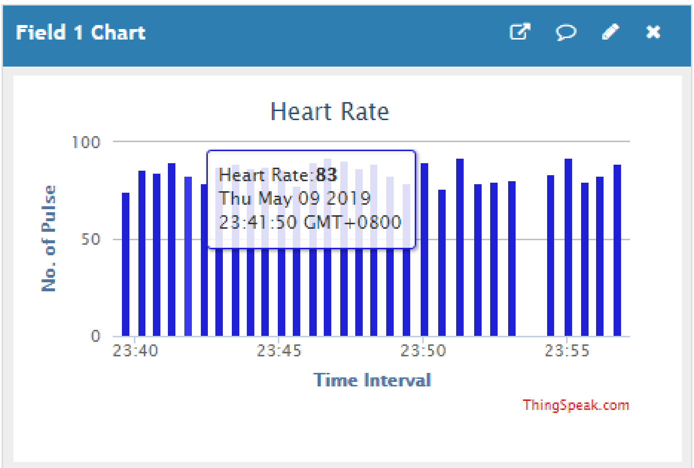


Figure 30: 5th data

6th data received at 23:42:23

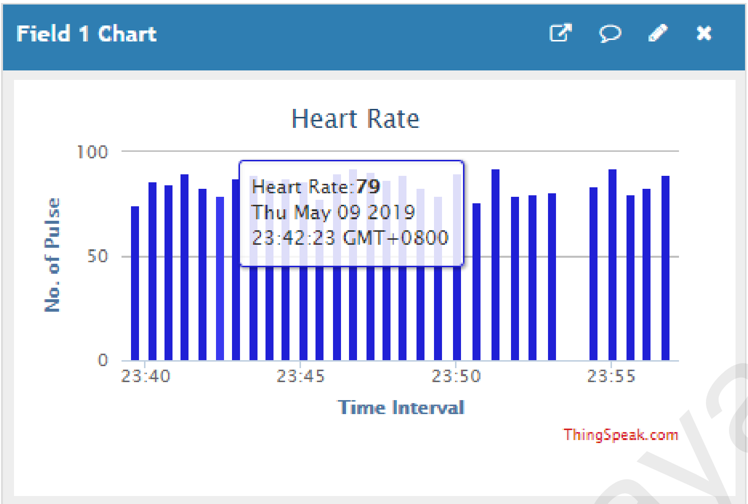


Figure 31: 6th data

7th data received at 23:42:55

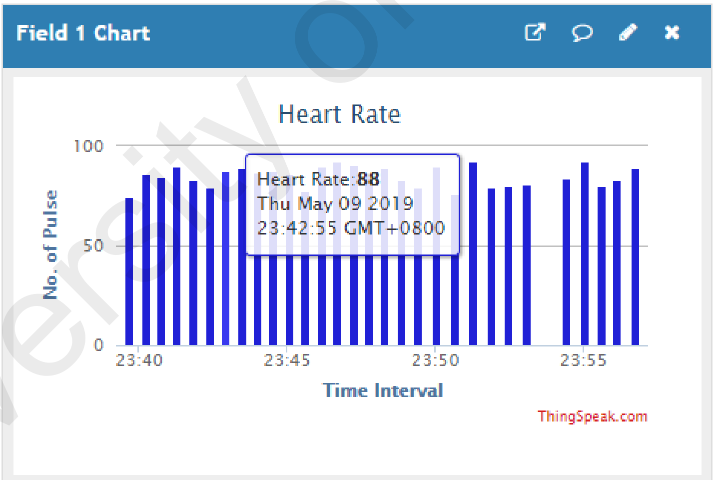


Figure 32: 7th data

8th data received at 23:43:28

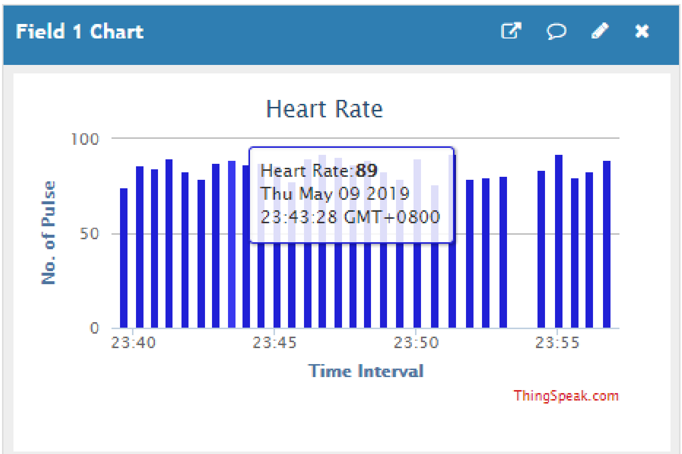


Figure 111: 8th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
3rd Data	23:40:45	-
4th Data	23:41:17	32
5th Data	23:41:50	33
6th Data	23:42:23	33
7th Data	23:42:55	32
8th Data	23:43:28	33
Average Delay in between each data		(32+33+33+32+33)/5 = 32.6

Table 12: Summary table of average data delay for pulse sensor

As the table shows that the average time delay for each data in heart rate sensor is 32.6 seconds at IPS.

Temperature Data

3rd data received at 23:41:01

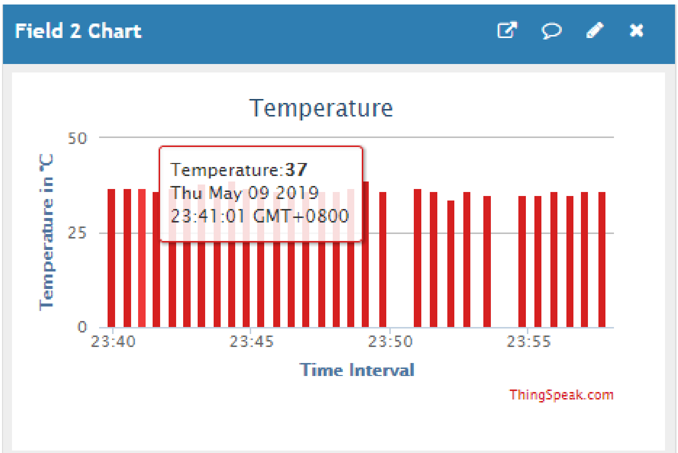


Figure 33: 3rd data

4th data received at 23:41:34

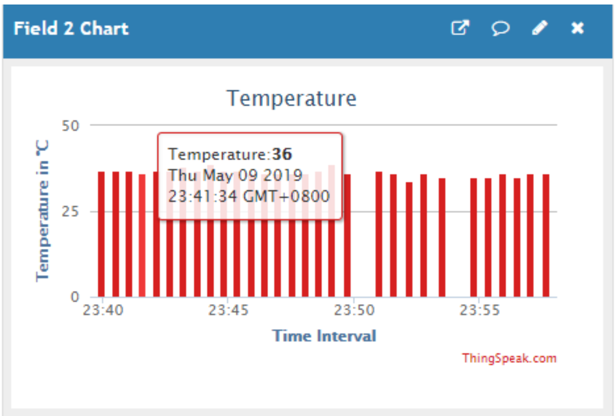


Figure 34: 4th data

5th data received at 23:42:07

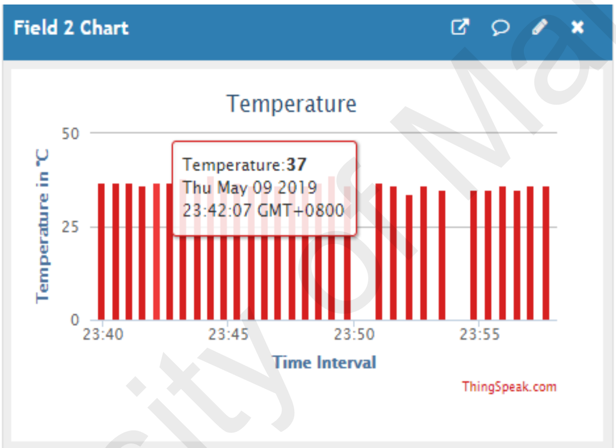


Figure 35: 5th data

6th data received at 23:42:39

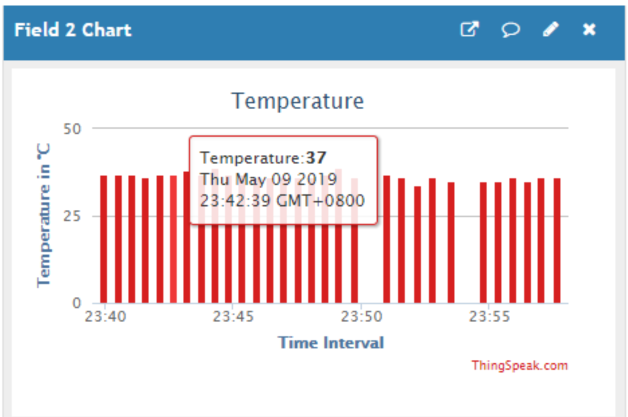


Figure 36: 6th data

7th data received at 23:43:11

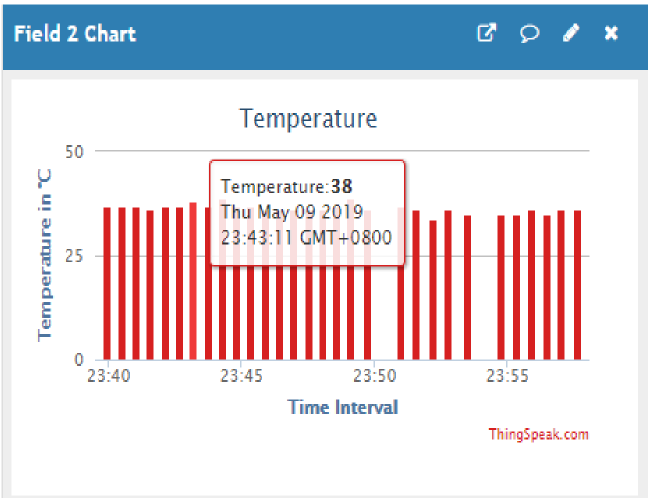


Figure 37: 7th data

8th data received at 23:43:44

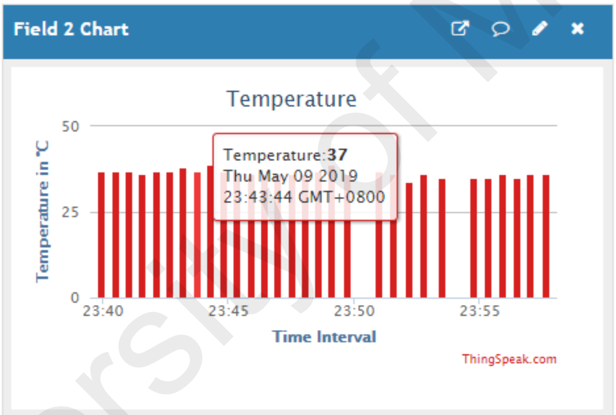


Figure 38: 8th data

Summary:

Data No.	Received Time	Time Delay between Each Data (in Seconds)
3rd Data	23:41:01	-
4th Data	23:41:34	33
5th Data	23:42:07	33
6th Data	23:42:39	32
7th Data	23:43:11	32

8th Data	23:43:44	33
Average Delay in between each data		(33+33+32+32+33)/5 = 32.6

Table 13: Summary table of average delay for the temperature sensor

As the table shows, the average time delay in between each data of temperature sensor is 32.6 seconds at IPS.

Time Delay in between both sensors for sending same data:

Data No.	**Data Receiving time for Pulse Sensor**	**Data Receiving time for Temperature Sensor**	**Time Delay in between Each Sensor Data (in Seconds)**
12th Data	23:40:45	23:41:01	16
13th Data	23:41:17	23:41:34	17
14th Data	23:41:50	23:42:07	17
15th Data	23:42:23	23:42:39	16
16th Data	23:42:55	23:43:11	16
17th Data	23:43:28	23:43:44	16
Average time delay in between both sensors' data			16.3333

Table 14: Summary table of average delay in between both sensors

As this table show, the average time delay in between both sensor data is 16.3 seconds at IPS.

Overall Findings from these Test:

Test No.	Download Rate (Mbps)	Upload Rate (Mbps)	Mobile Signal Strength	Mobile Network Type	Average Delay for Pulse Rate Data	Average Delay for Temper -ature Data	Average Delay in between both sensor's Data	No.of uneven delay (Heart Rate: Tempertu re)
1	23.48	20.43	-99dBm 7 asu	HSPA	38.2	37.6	56.5	4:3
2	10.65	5.31	-93dBm 10 asu	HSPA	36.2	35.8	54.167	4:5
3	27.88	1.36	-85dBm 14 asu	HSPA+	37.8	38.4	19	4:3
4	70.25	58.20	-85dBm 14 asu	HSPA	36.8	37	55	5:4
5	55.17	29.63	-85dBm 14 asu	HSPA+	37.6	37.6	18.833	3:4
6	92.71	54.45	-85dBm 14 asu	HSPA+	32.6	32.6	16.333	1:2

Table 15: Summary table of overall findings from the 6 tests

Analyzing the Overall Findings:

When mobile phone shows H symbol, it means that it has High Speed Packet Access (HSPA) and when H+ is displayed at the mobile phone, it means Evolved High Speed Packet Access (HSPA+). Basically H+ / HSPA+ provides faster network than H / HSPA; though H and H+ comes under 3G networks.

Mobile Signal Strength is measured in decibel-mill watts (dBm) and generally ranges from -50dBm to -110dBm. At the base of the Signal Tower the dBm is considered as 0 where the signal is very strong and further the distance increases from the tower the signal gets weaker. Since all the signal strength values are in minus the lower the dBm number gets the stronger the signal is. (Refer the below image).

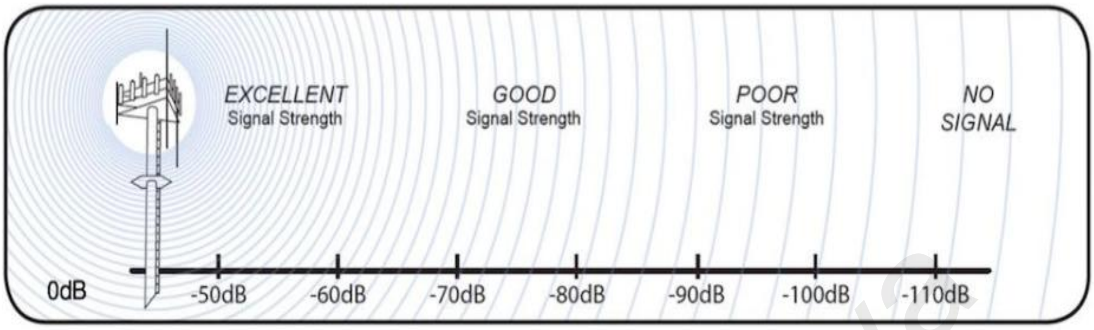


Figure 39: Scales of Mobile Signal Strength in dB. Image taken from Signalbooster (2019).

Arbitrary Strength Unit (ASU) is an integer value proportional to the Mobile Signal Strength. dBm = -113 + (2*ASU) is the equation for dBm to ASU conversions. Therefore the higher the ASU number gets the better the signal is.

Therefore the connection in between the dBm and ASU is; the lesser values of dBm give the higher values of ASU.

HSPA and HSPA+ don’t depend on these values and they may change individually without depending on the dBm and ASU values. E.g.: test 3 and 4, where same dBm and ASU but different network type.

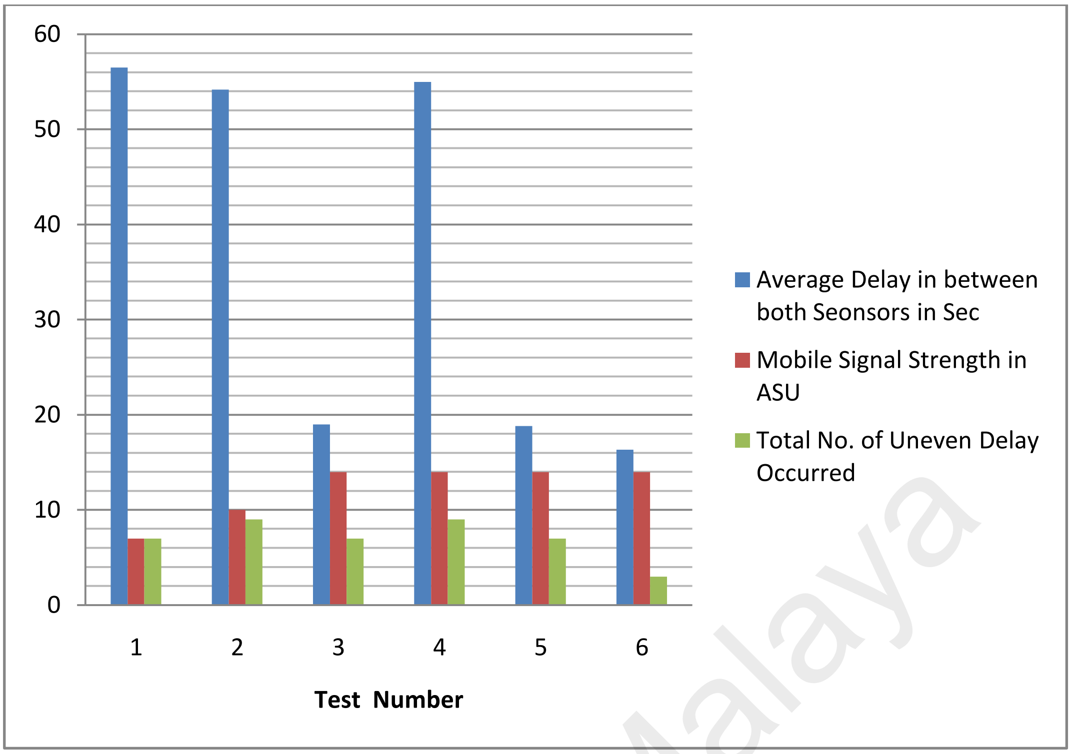


Figure 40 : Analyzing Chart of the Results

Average delay in between both sensors depends on Mobile Signal Strength as well as Network Type.

- In test 1, ASU = 7 and having HSPA network with 56.5 sec delay.
- In test 2, ASU = 10 and having HSPA network with 54.17 sec delay
- In test 4, ASU = 14 and having HSPA network with 55 sec delay.

Above 3 tests show that with the same network (HSPA), but different Mobile Signal Strength. When the network strength gets high the delay time must be reduced. This concept is OK to test 1 and 2 but test 3 deviates from this concept (must be less than 54.17 sec).

- In test 3, ASU = 14 and having HSPA+ network with 19 sec delay
- In test 5, ASU = 14 and having HSPA+ network with 18.83 sec delay
- In test 6, ASU = 14 and having HSPA+ network with 16.3 sec delay.

These 3 tests show HSPA+ with 14 ASU in all but having some slight different in between test 3 and 5 but 6 shows some more. But compare to HSPA, these HSPA+ networks directly impact on reducing the delay time in a huge amount. Unfortunately all the HSPA+ networks provide same 14 ASU values in the tests which does not allow predicting how the changes in ASU in HSPA+ network impacts on time delay.

Only in the 6th test, number of uneven delay occurred was reduced hugely under HSPA+ network and ASU = 14 but from the rest of the tests it is unable to predict the correlation between the network strength and type along with the uneven delays occurred since those values don't have any connections among them.

Likewise the Wi-Fi doesn't have any impacts on the results of this system. All the wi-fi values do not have any correlations between the mobile network or mobile signal strength or uneven delays.

The average values of pulse sensor and temperature data were useful to make sure that the data delay were even among them. But the both average values come in between 32 – 38 seconds in all tests; though the network is HSPA or HSPA+. Therefore network strength or network type doesn't have any impacts on the delays of each sensor individually.

CHAPTER 5: DISCUSSION

Entire test locations were situated within the University of Malaya's premises.

UM-WiFi was used during all these tests as a common internet provider.

Personal 3G mobile phone and DiGi sim card were used for Mobile Hotspot purpose.

Factors which impact the mobile signal strength such as heavy rain, closed area, several users of same network in one place, etc… were observed.

Wi-Fi Speed OR Mobile Signal Strength OR both factors were not stable during the entire tests which were unable to screen shot since they were frequently changing. Therefore average values of them were considered. These values were obtained by simply observing for values which comes frequently over a certain time period (10 – 15 minutes)

The biggest advantage of this system is; it is customizable up to 8 sensors in the circuit as well as at the IoT platform. Since this IoT system is free of charge, its functions are less with delays in data transmission. Paid version can provide real time data transmissions. (Underlined in red at the signup page). For testing the network strength and type, in which the IoT system works efficiently; this type of prototype is acceptable. Since nowadays all of us carry our own smart phone with us in most places; initiating the IoT based monitoring system is great. And further if not depending on the own hotspot, to initiate the system the unknown Wi-Fi's need to be accessed which reduces the integrity of the digital medical data. In theIoT based health monitoring systems, the network which is directly deal with the system (mobile hotspot in this case) play the major role in data delays and the other networks (UM Wi-Fi) which come along the system don't have any impacts.

Therefore the better the direct network to the system (mobile phone's network); the system works efficiently with less data delay. And if the network (mobile phone) does not drop from the average network strength measured before initiating the tests; there will be less uneven delays occur.

There were no relationships found in between the uneven delays occurred during the data transmission at the IoT platform. Therefore such uneven delay data areas were neglected and even data delay areas (may occur before or after such uneven data delay) were chosen for data delay analyzing.

It was observed that the number of uneven delays occurred more frequently whenever the mobile signal strength is gone below the average value of that particular environment.

Serial Monitor displays the senor values with decimal numbers but while displaying the same value at the ThingSpeak™ IoT Platform, the numbers were rounded off. i.e.: if the serial monitor shows 36.4℃, then at ThingSpeak™ it was displayed as 36℃. Likewise 35.8℃ was displayed as 36℃. This was done by the server settings since numbers with decimal need more capacity than numbers without decimals.

Since only one Arduino Wemos board is used for this entire system which is responsible for searching mobile hotspot and initiating the system, gathering sensor data, displaying them at the LCD screen and the Serial Monitor and sending them to ThingSpeak™ IoT platform; needs some delay among its major functions in order to handle all the data without missing any due to its high load of functions. Therefore the data in IoT server is displayed after 15 second time intervals in each sensor respectively.

LCD and Serial Monitor shows the same data at the same time but there were some unexpected data delay occurred at the IoT server sometimes due to the dropping of the Mobile Network Strength from its average values.

CHAPTER 6: CONCLUSION

A standard IoT based wireless health monitoring system has been developed and the performance has been evaluated in various environments. As the results show that whenever the faster Network Type (HSPA+ > HSPA) and the stronger Signal Strength (higher ASU numbers) was obtained; the less time delay in data transmission were achieved which is more important for such health monitoring systems.

Therefore, as some suggestions for improve the system performance are; applying high speed Network Types (i.e.: 3G > 4G) and make sure that the Signal Strength is not getting weaker due to factors like Environmental Conditions (i.e.: closed area reduces the signal strength), User Conditions (i.e.: more users in a particular area which makes single user's data transmission slower), Weather Conditions (i.e.: heavy rain which may weakens the signal strength), Distance between the Signal Tower and the User Point (Distance $\propto 1/\text{Signal Strength}$).

CHAPTER 7: FUTURE WORKS

- Threshold values for the vital signs can be set and Alarm systems can be added at the healthcare providers' station; so if the vital sign of a patient go below or above the threshold range, medical staffs can be alerted via the raising alarm and soon they can send immediate medical assistance to the spot.
- Much accurate value sensors can be used for this kind of systems which provide the less noise rate.
- Apart from using the personal mobile phone as a network provider, along with the network services the sensors attached with the phone also can be used such as accelerometer, gyroscope, GPS location services, finger print sensor, proximity sensor, etc… to get more vital signs, and additional data and reduce the number of body sensors.
- Since this system is customizable (maximum 8 sensors can be used) and the IoT platform allows to have 8 sensor data (8 fieds can be created in a single channel and 4 such channels can be created) more sensors can be attached with this system for getting more vital signs.
- To reduce more time delay in the IoT server and get real time values, to get mote storage space and more faster data transmission; paid version of the IoT platform can be used.
- Make more wearable sensors like wrist bands or combine the sensors along with the clothes such as smart t-shirt, smart hat, etc… better mobile networks provides can be used like 4G providers.
- Additional tests need to be carried out in many more environments and focusing on other environmental factors such as environmental temperature, whether condition, humidity, etc… and their contributions to the data delay.

- Since the entire system needs a battery os similar power sources; conventional power sources such as solar energy with additional backup batteries can be applied as the main power source for the system.
- Lastly, since the entire patient's data are handled digitally; the integrity, security, privacy, ect… of the medical data need to be confirmed.

REFERENCES

Arduino - Products . 2019. Arduino - Products . [ONLINE] Available at: https://www.arduino.cc/en/main/products. [Accessed 18 May 2019].

Arduino Stack Exchange. 2019. I2C LCD Serial Interface Board not displaying text (wrong pins?) - Arduino Stack Exchange. [ONLINE] Available at: https://arduino.stackexchange.com/questions/6782/i2c-lcd-serial-interface-board-not-displaying-text-wrong-pins. [Accessed 12 May 2019].

Breadboard \ Wiring. 2019. Breadboard \ Wiring. [ONLINE] Available at: http://wiring.org.co/learning/tutorials/breadboard/. [Accessed 18 May 2019].

C. C. Lin, C. Y. Yang, Z. Zhou, and S. Wu, "Intelligent health monitoring system based on smart clothing," International Journal of Distributed Sensor Networks, vol. 14, no. 8, p. 1550147718794318, 2018.

Doukas, C., & Maglogiannis, I. (2012, July). 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.

Doukas, C., & Maglogiannis, I. (2012, July). 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.

ElProCus - Electronic Projects for Engineering Students. 2019. Different types of Arduino Boards Used By Engineering Stundents. [ONLINE] Available at: https://www.elprocus.com/different-types-of-arduino-boards/. [Accessed 8 May 2019].

F. Jimenez and R. Torres, "Building an IoT-aware healthcare monitoring system," in 2015 34th International Conference of the Chilean Computer Science Society (SCCC), 2015: IEEE, pp. 1-4.

F. Nasri and A. Mtibaa, "Smart mobile healthcare system based on WBSN and 5G," IJACSA) International Journal of Advanced Computer Science and Applications, vol. 8, no. 10, 2017.

Faranux Electronics. 2019. OTA WeMos D1 CH340 WiFi Development Board ESP8266 ESP-12E For Arduino BRD31 - Faranux Electronics. [ONLINE] Available at: https://www.faranux.com/product/ota-wemos-d1-ch340-wifi-development-board-esp8266-esp-12e-arduino/. [Accessed 10 May 2019].

Gia, T. N., Rahmani, A. M., Westerlund, T., Liljeberg, P., & Tenhunen, H. (2015, April). Fault tolerant and scalable IoT-based architecture for health monitoring. In 2015 IEEE Sensors Applications Symposium (SAS) (pp. 1-6). IEEE.

Gia, T. N., Thanigaivelan, N. K., Rahmani, A. M., Westerlund, T., Liljeberg, P., & Tenhunen, H. (2014, October). Customizing 6LoWPAN networks towards Internet-of-Things based ubiquitous healthcare systems. In 2014 NORCHIP (pp. 1-6). IEEE.

GitHub. 2019. GitHub - witnessmenow/ESP8266-4051-Multiplexer-Example: An example showing how to use a 4051 multiplexer with an esp8266 to connect up to 8 analog sensors.. [ONLINE] Available at: https://github.com/witnessmenow/ESP8266-4051-Multiplexer-Example. [Accessed 10 May 2019].

H. Basanta, Y.-P. Huang, and T.-T. Lee, "Intuitive IoT-based H2U healthcare system for elderly people," in 2016 IEEE 13th International Conference on Networking, Sensing, and Control (ICNSC), 2016: IEEE, pp. 1-6.

Home - STMicroelectronics. 2019. Home - STMicroelectronics. [ONLINE] Available at: https://www.st.com/. [Accessed 21 May 2019].

Jimenez, F., & Torres, R. (2015, November). Building an IoT-aware healthcare monitoring system. In 2015 34th International Conference of the Chilean Computer Science Society (SCCC) (pp. 1-4). IEEE.

K. M. Alajel, K. B. Yosuf, A. R. Ramli, and E. S. Ahmed, "Remote electrocardiogram monitoring based on the internet," kmitl sci. J, vol. 5, no. 2, pp. 493-501, 2005.

K. Ullah, M. A. Shah, and S. Zhang, "Effective ways to use Internet of Things in the field of medical and smart health care," in Intelligent Systems Engineering (ICISE), 2016 International Conference on, 2016, pp. 372-379: IEEE.

L. Catarinucci et al., "An IoT-aware architecture for smart healthcare systems," vol. 2, no. 6, pp. 515-526, 2015.

Learn More - ThingSpeak IoT. 2019. Learn More - ThingSpeak IoT. [ONLINE] Available at: https://thingspeak.com/pages/learn_more. [Accessed 10 May 2019].

M. S. S. Kenganal and P. J. I. J. o. I. R. Rengaprabhu, "Real Time Wireless Patient Monitoring System Based On IOT," vol. 2, no. 13, 2016.

MakeUseOf. 2019. EDGE, 3G, H+, Etc: What Are All These Mobile Networks?. [ONLINE] Available at: https://www.makeuseof.com/tag/edge-3g-h-etc-mobile-networks/. [Accessed 20 May 2019].

P. Gope and T. J. I. s. j. Hwang, "BSN-Care: A secure IoT-based modern healthcare system using body sensor network," vol. 16, no. 5, pp. 1368-1376, 2015.

P. Gupta, D. Agrawal, J. Chhabra, and P. K. Dhir, "IoT based smart healthcare kit," in Computational Techniques in Information and Communication Technologies (ICCTICT), 2016 International Conference on, 2016, pp. 237-242: IEEE.

P. P. Ray, "Home Health Hub Internet of Things (H 3 IoT): an architectural framework for monitoring health of elderly people," in 2014 International Conference on Science Engineering and Management Research (ICSEMR), 2014: IEEE, pp. 1-3.

Pulse Sensor Pinout, Configuration & How Pulse Sensor Works. 2019. Pulse Sensor Pinout, Configuration & How Pulse Sensor Works. [ONLINE] Available at: https://components101.com/sensors/pulse-sensor. [Accessed 7 May 2019].

R. Kodali, G. Swamy, and L. Boppana, An implementation of IoT for healthcare. 2015, pp. 411-416.

SignalBooster.com. 2019. How to measure signal strength in Decibels on your cell phone? . [ONLINE] Available at: https://www.signalbooster.com/blogs/news/how-to-measure-signal-strength-in-decibels-on-your-cell-phone. [Accessed 20 May 2019].

T. Wu, F. Wu, J.-M. Redoute, and M. R. J. I. A. Yuce, "An autonomous wireless body area network implementation towards IoT connected healthcare applications," vol. 5, pp. 11413-11422, 2017.

Temperature Analog Sensor Module LM35D | QQ Online Trading. 2019. Temperature Analog Sensor Module LM35D | QQ Online Trading. [ONLINE] Available at: http://qqtrading.com.my/temperature-analog-sensor-module-lm35d?search=temperature%20sensor. [Accessed 5 May 2019].

V. M. Rohokale, N. R. Prasad, and R. Prasad, "A cooperative Internet of Things (IoT) for rural healthcare monitoring and control," in 2011 2nd International Conference on Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE), 2011, pp. 1-6: IEEE.

Yeh, K. H. (2016). A secure IoT-based healthcare system with body sensor networks. IEEE Access, 4, 10288-10299.