REMOTE PATIENT MONITORING USING HEALTH BANDS

WITH ACTIVITY LEVEL PRESCRIPTION

by

PRANAY SHIROLKAR

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE AND ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2016

Acknowledgements

This thesis is a blend of motivation and insights from Prof. David Levine, my Guru, mentor and supervisor. I would like to thank Prof. David Levine with all my heart for encouraging me, motivating me and tracking my progress throughout the work that was being done. It would not be possible without your belief in me and tons of anecdotes from your experience. Thank you again for enlightening me through the path of my career and all the wisdom that I have gained under your supervision.

I would like to thank all my committee members, for their patience, faith and confidence in me. I thank Dr. Khosrow Behbehani for his fruitful insights and guidance for the fruition of this project and systematically guiding the way and motivation. I thank Dr. Christoph Csallner to provide me insights on the aspects of software engineering, motivating to pursue something that I believe in and identifying the potential in me.

I thank my parents, Mr. Girish Shirolkar and Mrs. Sumati Shirolkar, without their constant care, hope, faith, motivation, and confidence in me I cannot imagine the completion of this thesis.

I thank Purva Sugandhi for always being there and having a very patient and perseverant effort at listening and constantly motivating me towards achieving my goals.

I thank all the people, my friends, family, relatives, classmates, colleagues, acquaintances whom I have not included here specifically, but they have played an important role in the pursuit of my higher studies and goals.

November 3, 2016

Abstract

REMOTE PATIENT MONITORING USING HEALTH BANDS

WITH ACTIVITY LEVEL PRESCRIPTION

Pranay Shirolkar, MS

The University of Texas at Arlington, 2016

Supervising Professor: David Levine

With the advent of new commercially available consumer grade fitness and health devices, it is now possible and very common for users to obtain, store, share and learn about some of their important physiological metrics such as steps taken, heart rate, quality of sleep and skin temperature. For devices with this wearable technology, it is common to find these sensors embedded in a smart watch, or dedicated wearable wrist bands such that among other functionalities of a wearable device, it is capable of smartly assisting users about their activity levels by leveraging the fact that these devices can be, and are typically, worn by people for prolonged periods of time.

This new connected wearable technology has a great potential for physicians to be able to monitor and regulate their patients' activity levels. There exist many software applications and complex Wireless Body Area Network (WBAN) based solutions for remote patient monitoring but what has been lacking is a solution for physicians, especially exercise physiologists, to automate and convey appropriate training levels and feedback in a usable manner. This work proposes a software framework that enables users to know their prescribed level of exercise intensity level and then record their exercise session and securely transmit it wirelessly to a centralized data-store where physiologists will have access to it.

Table of Contents

List of Illustrations

Chapter 1

Introduction

Fitness devices that are available in the consumer market are capable of measuring health/physiological metrics that are capable of aiding doctors in gaining more insight to the patient's overall health. Effects of exercises have been studied and it has been established that people (of all ages) who regularly maintain a physically active lifestyle by doing regular exercise have a very positive impact on their health and people who do not, are at a substantially high risk of chronic illnesses [1, 2]. Many doctors, exercise physiologists and physicians have been incorporating physical exercises as a part of their treatment plan along with their prescription of medicine. There are some specific cases where the practice of prescribing exercise is more suitable or required and is also very effective in the recovery, for example, in patients with fractures that render a part of the body immobile for several months, patients with traumatic brain injury (TBI) who have reduced or impaired mobility and need regular exercise to gain mobility and especially with people of age. Patients who spend a relatively large amount of time in recovery, rehabilitation doing exercise and physical training with incremental intensity level often need to continue a specific amount of exercises consistently for months without supervision as well as in order to have a successful recovery.

Technology that is available today has the capability to allow doctors to gain much deeper understanding of the patient's health. However, there exists no way for doctors to get immediate, reliable and consistent feedback about the patient's physical activity. When there is a crucial need of such information, doctors have very little choice how to get access to the health data from patient's everyday health. Specifically, in cases where a specific level of daily physical activity is crucial to be maintained for recovery and where any delay or gap in the exercises can lead to irrecoverable impairment, it becomes important for the clinician to get continuous feedback from the patients and to set a customized goal for the patient after analyzing the patient's previous activity levels.

Background

Physical fitness is one of the prime metrics towards a healthy life. These days, More and more people are getting used to live a very sedentary lifestyle. People are already getting very concerned about the level of their physical activity and their day to day health tracking. Consumer grade commercially available mobile devices such as Apple iPhones, Android Smartphones, other accelerometer pods and wearables have been available in the market for many years. These devices monitor various health metrics using a number of sensors, one such example is an accelerometer. An accelerometer is a device that measures the amount of acceleration in each of the three orthogonal axes. This allows the device to do calculations on physical movements such as step counts, walking distance and running distance that are important health metrics.

Big technological companies have already entered this market with interest in this new wearable technology and are developing sophisticated software and hardware devices that allow users to track, maintain and share their fitness activities [29].

One promising possibility is the application of analytics to the data collected by these fitness devices. Each person will have a history of his important physiological metrics obtained from their monitoring device(s). This could be very helpful in diagnosing and researching some causes behind some common chronic illnesses. Having this knowledge will also allow researchers to map certain parameters of commonality within these datasets.

Computer machine learning is a technique of allowing a computer to learn to derive results by training the system using existing or previous related data. Access to these large sets of data coupled with technology such as big data and cloud computing is one aspect of the future work of this proposal.

According to a survey conducted in 2014 [29] 42% of the people said that using the wearable fitness devices to monitor progress has the potential of improving the athletic ability of an average person. This also depicts the popularity and acceptance of the fitness devices that are currently available in different segments of the consumer market. The types of wearable devices can be classified as wearable watches or bands, foot pods (a device that attaches to your shoe), belt pod, smart glasses, smart phones, smart textile that has embedded sensors like Ralph Lauren's Polo Tech Shirt and OMSignal's smart fabric for women [31]. These wearable devices have sensors attached to it such as accelerometers, gyrometers, barometers and optical heart rate and skin temperature monitors that are discussed in much more detail in the following sections.

The easy availability and low prices of these devices can be very helpful for patients who will be greatly benefited by the use of such devices for tele-medicine and tele-exercise that will eliminate the need to be physically present with a doctor in attendance. Further, for doctors, the tele-monitoring of their patients' health metrics will provide granular and insightful information to the doctors that has not been possible before. Even though these devices are readily available and there are many software programs that already connect and share the data that comes out of these devices, there is no tool or software or way where a doctor, if he wants, would be able to keep track of his own patients using a centralized system.

The key need here is for a software platform that is capable of communicating with all these variety of devices but still being able to intake data in similar terms that is useful for the doctors to monitor their patient's health and activity remotely using apps and a centralized cloud technology. This can be done for one ecosystem of software / hardware technology. However, this means that the platform should be capable of delivering this streamlined view to the doctor's from different types of devices which is usually not possible and different manufacturers' devices don't work and communicate in the same way. Furthermore, a doctor's ability to set the amount of intensity of exercise that their patients should be a great tool because that serves as a prescription and provides a feedback system for the patient. This means that if a software system is openly made available for patients and doctors to use where they can securely store, share and manage their health data, it will serve as a great tool for the upcoming community.

Related Work

There exist many software systems that allow a similar type of functionality where they can share their metrics with others. MyChart from Epic Systems is an application that allows patients and doctors to share a number of things in common. It allows doctors to create prescriptions for medicines that they are supposed to take. The app on iPhone also has the capability to share health data such as weight, height, steps [33]. Appointments and reminders with a doctor is another feature of this application. It has a mobile app component and a doctor's app that works cohesively. This app is being used in many healthcare providers across the United States, such as UT Southwestern Medical Center [34].

Microsoft's HealthVault is the most closely related software service that allows the facility to organize health records and all other medical issues, complexities and health metrics within their software system. Some examples of such metrics are the medications, health history, allergies and blood pressure. Microsoft HealthVault provides integration services with a number of other device manufacturers that also allow it to incorporate the wearable fitness tracker device data into its system. HealthVault works with healthcare providers and professionals in order to share this data with your doctors. This work differs from Microsoft HealthVault by directly accessing the health metric data from the wearable device. Another important difference is that this work proposes a software framework that aims to assist the physiologists to a further step by allowing them to add an exercise prescription on goal exercise intensity level within the software which is accessible to the patients while they perform exercise. Microsoft HealthVault is a central data store of all the health metrics that can be stored, progressing further towards the digitization of medical records. However, this work proposes providing access to wearable devices' data to the doctor. The software application proposed could be one of the many other applications that transmit data to the Microsoft HealthVault for further storage and sharing.

Wireless Body Area Networks (WBAN) is a wireless network of wearable computing devices [43]. These devices can be of different types: embedded or implanted within the body, or attached to the body on the surface or skin and may also be the ones that one can carry around with one in different places for example in or on a shoe, on the waist belt, in the purse or around neck. IEEE standard 802.15.6 has been put in place that contains standard policies for devices, software and networks which aim to work with collecting such data [44]. It lays out in great detail the security protocols and policies to maintain the integrity, privacy, confidentiality, etc. of the sensitive data. Wireless Personal Area Networks (WPAN) is the parent technology of the WBAN where it initially started and then it the body area networks was branched off from the personal area network in

1995 [43]. The term “BAN” was introduced several years later after this separation took place [43].

One of the many works is by Ogunduyile et al. [42] which propose an Arduino based WBAN network to provide a system of ubiquitous healthcare monitoring. They have proposed a software solution that communicates with its hardware component to provide metrics by using data from sensors such as Triaxial Accelerometer, SpO2 sensors, and location sensors. Ogunduyile et al. [42] have proposed a live prototype of their system called the SOWBAN architecture that is divided into three layers – physiological sensors in the first layer, a personal server in the second and the third one consists of all the health care systems and services.

Apple has a relatively similar program Research Kit [3] that aims to provide medical researchers a platform to get health data from iPhones and Apple Watches towards their study of health metrics. Apple’s framework Health Kit is what backs the Apple family of devices in storing and managing of health metric data. This is already a very good source of historical information of people who have been storing it in the system. A Google equivalent of this framework is Google Fit. Apple Health data is not exposed to a user except by creating an app on the device itself. A lot of other technology exists that allow to enrich the health data on Apple such as iHealth [35].

On the other hand, Google’s Google Fit data store is available externally through a REST API [30], which is helpful for gathering data independent of any mobile app created specifically for that function as opposed to the case with Apple Health. [36] and [37].

Additionally, concepts of measuring, storing and sharing the fitness and health data, there are other technological advancements that have been in this domain that map to the remote monitoring of patients. Zephyr Life Home Remote Patient Monitoring [38] which is a subsidiary of Medtronic, also have their own remote patient monitoring system [39] that is separate from the Zephyr system. Both of these systems have a function of keeping the patient's vitals connected through a network to their doctors and hospitals. Medtronic has also incorporated devices that have remote monitoring capabilities by using an external externally powered unit that will transmit the data from sensor to doctors [40], [41].

ZephyrLIFE Home remote patient monitoring system is a complex system built using a combination of their products that are already manufactured and used in the medical and clinical domain. It uses ancillary devices to monitor vitals such as blood glucose, blood pressure, body temperature, body weight, and oxygen saturation. They have a software platform that runs on Android based hand held devices such as a smart phone or tablet called HealthHub. This transmits data wirelessly to their web application software component that can then be reviewed by a clinician [38]. This system is a complex integration of all the sensors and devices that work to move medical grade patient monitoring system to home.

There are independent applications from manufacturers of fitness devices such as Fitbit and Jawbone that have mobile application which will allow a person to share their health metrics with other peers such as friends and family.

Motivation

There had already been a widespread acceptance to the emergence of new software and hardware that is capable of measuring various kinds of health metrics. For example, phones have accelerometer and gyrometer that easily allow software to extract step count and moving distance data. The software systems such as Google Fit, Microsoft Health and Apple Health also provide great connectivity to different types of wearable health devices that are able to report even greater levels of reliability with their data because they are worn on the user's body. These devices are also equipped with some other useful sensors that are capable of measuring your skin temperature, skin resistance, sleep quality and heart rate.

Physicians who deal with patients that need regular amount and intensity of exercises over a long period of time as a part of their treatment were the people who could greatly benefit by these kinds of devices. The deficit of such a technological solution that would allow physicians to monitor and analyze that data was recognized during congress of this thesis' committee chairman, David Levine and researcher at Institute of Exercise and Environmental Medicine (Texas Health Presbyterian of Dallas), Rong Zhang.

Dr. Rong Zhang at The Institute of Exercise and Environmental Medicine works with fellow exercise physicians who employ chest-worn heart rate monitors to gather health data about his patients. Another big challenge about this system was that there is no way for the physicians to get feedback although the exercise data was being collected by the patients on their monitors on an everyday basis. The only way they are able to get access to that data is at a frequency of once a month when the patient comes with the device which then connects to physician's computer and transmits the stored data. This delay was identified as unworkable given that many times gaps in the sessions were not revealed until about a month later when the patients arrive with their devices.

Having being interested in the metrics of a person's health, after an initial study about the types of devices available and their programmability to be used with a custom app, this deficit of an intuitive technological solution, became the motivation of this work. Previous experience in fast paced software development towards interactive solutions to real life problems has always been a big driving force.

Given the amount of positive impact such a software system can have on society and a great potential of success, this work commenced with a goal of creating a software solution that is usable by physicians who are already using or are trying to get some kind of similar feedback about their patient's health data.

Goals of the thesis

This thesis proposes a software solution using key technologies such as cloud computing and wearable technologies that allow doctors to monitor patient's health and fitness metrics and data as calculated by the sensors on new wearable fitness tracker devices; and set specific exercise intensity goals or exercise prescriptions for patients. It proposes using a consumer grade, commercially available fitness device that can be worn around wrist to collect data from patients. A smartphone app that communicates with patient about their goals and allowing them to record an activity or exercise session. It will also be responsible for communicating with the band for collecting data and sending it for storage and processing to a server. A web interface is proposed for doctors or clinicians where they set goals for their patients and analyze and view the current data to further customize the goals.

The software systems that already capture this kind of user health data have already been implemented by big technological companies. The big players in this category are Apple Health, Google Fit and Microsoft Health. All of these software systems work in a similar fashion: they collect health data from a number of devices – primarily the user's phone, and then wearable devices like smart watches and health bands. However, the capability for the software to provide reliable feedback to the user is not yet present. Providing physicians with the access to such data to provide valuable feedback to the patient, simultaneously facilitating doctors to constantly monitor and provide feedback from their end is the goal of this thesis.

This work proposes an intuitive system of prescribing exercise intensity level as goals within the software platform. The doctor's web interface allows them to input and set a defined goal that is displayed on the patient's mobile application. This allows the patient to know a reference for their exercise goal and their goal for the exercise session. A graphical cue for this goal is also plotted when generating graphs for the activity levels for the doctor to observe on the web application.

There is great opportunity for the proposed software system solution to use this existing historical data about the patient that may be available from these proprietary software systems. This data can be processed and analyzed with the current data to employ data analytics to assist the physician while he looks at the raw data and figures.

Organization of the thesis

The thesis starts with Chapter 1 that comprises of introduction and background, motivation and goal of the project and the relevant technologies, history and trends. Chapter 2 is titled Means and Methods and specifically includes the technologies used in the development of this thesis project. This includes discussion of various types of health devices available, types of sensors employed with a brief discussion on each of them, various health metrics that these health devices measure. The last subsection of Chapter 2 includes the specific technical details of the software implementation of the project that is divided up in three parts: a web application for the doctor, a mobile application for patients, and backend services to manage the data and processing of the system. Chapter 3 discusses the results of the project, the user interfaces developed and the functional overview of the software developed. Chapter 4 discusses the thesis, project and the feedback from Marcel Turner who is an exercise physiologist, and other committee memebers apart from other general conclusions. Lastly, Chapter 5 outlines the future work for the project discussed in this thesis.

Chapter 2

Means and Methods

Health Devices Available & Sensor Capabilities

At the time of writing this thesis (Fall 2016), many different types of wearable health devices are available in the market. They can be classified in various types of categories. Based on where they are worn: Wrist-based watch type, chest-strap type devices, waist and foot worn devices. A variety of these devices were analyzed and studied on a number of factors such as type of programmability and data flow, cost, types of sensors and data available, and ease of use for writing this thesis. One other type of classification can be done on the basis of purpose of manufacturing of the device. There are research oriented medical grade devices and consumer grade devices which are comparatively cheaper compared to the medical grade devices.

A combination of a number of different types of sensors are usually found to be employed in these devices.

Sensors

Accelerometer

Accelerometer is a device that detects motion by measuring acceleration [4]. Typically, the accelerometers employed are tri-axial, i.e. they report simultaneous values in three orthogonal directions. This is helpful in measuring step count, walking or running distance and to indicate periods of restful and light sleep [6].

Gyrometer

It is a device that uses Earth's gravitational force to determine orientation. This coupled with accelerometer make a very powerful motion detector because the accelerometer lacks orientation sensing capability and this augments the metrics that are obtained from an accelerometer [7, 6 and 5].

Photo-plethysmography Sensor (Optical Heart Rate Monitor)

An optical heart rate monitor that works to determine heart rate by emitting light and measuring the amount that was absorbed. Since the blood absorbs light, difference in the absorption between two beats is measured by high frequency sensors to measure heart rate [8].

Optical Thermometer

Small infrared sensor that is capable of measuring skin temperature.

Global Positioning System Sensor (GPS)

Provides pin pointed location in longitude and latitude by using information from the satellites orbiting the Earth. GPS is not very common most of wearable devices because of one big drawback: it is power hungry. And since mostly, streamlined design and battery performance is more important than a location metric, one can easily find variation of devices even from same manufacturers of fitness devices that are equipped with or without GPS.

Ambient Light Sensor

This sensor detects light in a similar way as a human eye.

Electro-dermal Activity or Galvanic Skin Response Sensor

Bio-impedance sensors works by measuring the resistance of skin to a tiny electric current [9]. This is considered indicative of “sympathetic nervous system arousal and to derive features related to stress, engagement, and excitement” [10].

Ultra Violet Light Sensor

This sensor is capable of detecting UV Radiation intensity using photo sensors specific to UV radiation spectrum. [11].

Microphone

A microphone can be used to provide voice commands to the device and to detect any noise while tracking sleep patterns.

Barometer

Barometer measures the change in air pressure to determine your height of elevation when you are climbing stairs of flight, or hiking for example. [6].

Research Grade Health Devices

Empatica E4 Band

This band is a medical grade research oriented health device manufactured primarily for clinicians who need to monitor their patient’s physiological metrics while the patients are being closely monitored for their activity. This band is capable of streaming sensor data to the app. The Empatica E4 band uses several of the sensors that include heart rate sensor, electro-dermal activity sensor, accelerometer and optical thermometer.

Consumer Grade Health Devices

Microsoft Band

Microsoft Band is the first generation smart band with smartwatch and activity tracker/fitness tracker features, created and developed by Microsoft [12]. This can be integrated with Windows Phone, Android and iPhones via a Bluetooth connection. It has a companion app called Microsoft Health (now called Microsoft Band on some devices) that works with the band to get fitness and health data from the band. The second generation of the same line was released about a year later in late 2015 and was named Microsoft Band 2.

It has almost all of the sensors discussed above that include accelerometer, gyrometer, optical heart rate sensor, GPS, microphone, ambient light sensor, barometer, galvanic skin response sensors, UV sensors and skin temperature sensors.

Polar H7 Chest Strap Band

Polar is a manufacturer of sports training computers known for developing the world's first wireless heart rate monitor [13]. Polar has a line of straps developed that go around the chest and communicate with a wrist band to display heart rate. Polar H7 is the first one to have a Bluetooth capability. Bluetooth capability allows flexibility in the number of devices that the chest strap band can be used with. This also allows devices that are not branded by Polar to use the heart rate readings that come out of the Polar H7 chest strap band.

Apple Watch

Apple Watch is a line of smartwatches developed by Apple Inc. It incorporates fitness tracking and health-oriented capabilities with integration to Apple's iPhone's operating system, iOS and other Apple products and services. [14] Apple watch communicates to devices using Bluetooth connectivity and was announced by CEO Tim Cook in September 2014, and released in April 2015. The Apple Watch quickly became the best-selling wearable in the world with 4.2 million watches sold [15].

Polar A360

This is Polar Electro's first wrist based heart rate monitoring wearable device that was released in October 2015 [16] and [17]. It is a waterproof band and allowes for Polar's customers who want to make a switch from the chest strap heart rate monitors to the wrist worn monitor.

Polar M600

Polar M600 is Polar's most recent advancement in the wearable technology. The much needed GPS functionality was added with the waterproof IPX8 rating that makes it usable while swimming [18]. Polar M600 has a better accuracy at detecting heart rate than is predecessor, Polar A360, because of the fact that they opted to use 6 LED sensors instead of the usual 2 for measuring heart rate. In addition, there exist ways to connect third party or Polar's chest strap bands if they really need pin pointed accuracy. A comparison of heart rate performance of M600 with a chest strap band has been shown on [18].

Fitbit Charge 2

Fitbit is a renowned manufacturer of fitness trackers and activity trackers. It has a large number and variety of fitness trackers with different types of capabilities. Fitbit Charge 2 is capable of heart rate monitoring, steps and calorie tracking, different types of workouts such as biking and yoga. Although it has most of the sensors that were discussed in section above, it lacks water resistance and GPS. [19].

Health Metrics

Heart Rate

Heart rate measurement has become an important metric of health. In the clinical domain, both physiological and psychological applications of the heart rate measurement have been found [20]. The variability in resting heart rate is often tied to detect illnesses such as diabetes and obesity early [66-67]. Resting heart rate indicates conditions about one's body. A high resting heart rate usually means that the heart is having to do more work to maintain blood flow across the body. In athletes and people who often perform cardiovascular activity, the resting heart rate is usually low. After a strenuous workout, the body's metabolism and heart work harder to bring back a state of homeostasis [21]. there may be other reasons for an increased resting heart rate. Apart from that, resting heart rate also depends on physical size of heart, body, fitness level, medication use, etc. [21].

Heart rate measurement during exercise, on the other hand, is an effective metric because it is an objective measure of the intensity of your exercise [22]. Apart from being informed about possibilities of undertraining and overtraining during workouts, being in an appropriate heart rate zone is also something that has known to achieve weight reducing goals, which is also why many treadmills have a heart rate chart on them [23].

Steps and Running

Walking is one of the most basic forms of expenditure of energy. It is also one of the most common and easily achievable forms of exercise, especially for middle aged and older people [24].

Most fitness devices today have come up with an estimated distance of your walking and distance travelled, as well as the number of flights climbed and the altitude gained using a combination of accelerometer, barometer and gyrometer. Some of the very common examples, as we discussed in an earlier section are - Apple iPhone, Apple Watch, Fitbit wrist bands, other smartphones and so on.

Many of these devices have their own, or third party developed software, to create and achieve everyday goals. More sophisticated software will also allow sharing your goals and challenges with your friends, creating a competitive environment towards sustaining a healthy life through the use of technology.

Calorie Expenditure

This is a very useful health metric that most all of the consumer market devices attempt to provide – the information of one's estimated calorie burn. This calculation is often based on incorporating the person's height and weight, age, and activities performed during the awake hours. Some devices also provide an estimated calorie burn of your asleep hours, for example, Microsoft Band.

Sleep Quality Tracking

There have been research studies [25-27], that have established that a restful sleep – that lasts for long enough time and that is uninterrupted, is essential towards maintaining a good health down the line. Effects of sleep deprivation have been directly correlated with stress, obesity and lack of concentration and cognitive ability [25-27].

Sleep quality is usually expressed by indicating the number of times a person was awake during the sleep and for at what time that observation was noted and the other hours of actual sleep. Sleep efficiency, as explained on Microsoft's health dashboard is defined as the ratio of time spent asleep to the time the total time spent in bed. Other quantifiable measures that the devices provide also include the amount of sleep split between restful sleep and light sleep. A restful sleep is a deeper, restorative phase of sleep and the light sleep is a relatively restless phase of sleep where one have not yet entered a deep sleep phase.

Exercise Session Tracking

Many of the devices that are now in the consumer market, such as the Microsoft Band and Fitbit Charge 2 attempt to recognize the type of activity that one is doing once one learns the exercise being done a few times [28]. Common patterns are often easy to recognize, such as walking, running, bicycling and sleeping. However, it is difficult for health bands to effectively measure the effectiveness and the pattern of more complex exercise sessions such as weight lifting, stretching, yoga, or some other kinds of cardiovascular and anaerobic exercises that will have different effects. This creates the need for setting a customized exercise session within the device's ecosystem that will then be capable of measuring the metrics at those sessions more effectively.

Validation of Accuracy and Reliability of the Devices Used

Knowing the practical precision and accuracy of the devices being used to propose this framework is of importance since these commercially available devices are not FDA approved. Also, according to a recent policy update from FDA that concerns and includes these low risk devices, will be kept away from the scrutiny and examination under the FD&C Act [58]. This makes it important to determine the accuracy of the devices that are going to be used to determine the various health metrics or vitals. It is expected that these devices will not be as accurate as the clinical grade devices that are used in clinics and hospitals. As an example, the optical heart rate monitoring or photo-plethysmography, the technique used to measure heart rate on devices that are worn on wrists, has its limitations of not being properly worn or not fit properly, movements during activity, sweat or moisture, and other physiological factors. [59]

Even the wrist worn devices may have their limitations, they have been tested and validated against the previously established to be more accurate devices, such as chest strap electrode based devices that were previously tested against the clinical grade standard EKG. We have reviewed several of the comparisons and comparative study papers that go through the accuracy of these various kinds of devices. The most significant of these is [60] which compares the 6 different brands of commercially available wrist worn heart rate monitors including Microsoft Band, Fitbit Charge HR and Basis Peak. The basis of validation was the Polar RS400 chest strap band that was previously validated against ECG as indicated in the discussion. The study participants were a variety of population in range of age 19 to 43 years that had male and female participants totaling 50. The protocol included continuous walking and running at various levels of intensity. As a conclusion of the study it was determined that the activity trackers provide an accurate measurement of heart rate during the activities as the protocols tested [60].

Apart from this paper setting the basis for use of similar devices for this proposal and experiment, there have been other comparative studies which provide information about the tests that have been done and this also provides us with a certain level of credibility of a larger variety of the devices. This is important in this project because we need the capability to subscribe to the sensor readings in a direct manner, and not all of the devices have the capability to do that. Some of the devices that we have been found to have that capability are: Microsoft Band, and Apple Watch. Most of the other bands don’t allow direct access of sensor readings but instead a processed set of data coming from their cloud APIs that will record general running, or other workout sessions, recorded using the band or their app’s interface. It is possible to use the data in that way as well. This will allow integrating more types of devices in this project than what is now possible. According to the comparative study performed and described in [61] Microsoft Band 2 was found to be the most accurate among the 4 devices that were tested in the same price range and they all use the same technique to detect heart rates. According to another comparison done by Shane Richmond in [62] Adidas miCoach and Garmin Forerunner 225 are the most accurate while noting the fact that none of them matched the accuracy achieved by a chest strap heart rate monitor, among other conclusions. However, as the author states in [62] “This experiment was not a scientific test but if it was, it would conclude that more research is needed”. However, about the same time, the scientific study done by Stahl et al. in [60] concludes that these wrist heart rate monitors are as good of detectors as the chest strap detectors.

Inferring from some of the other independent experiment done in [63] and an infographic in [64] and [65] it can be concluded that the wrist based sensors are somewhat accurate with some challenges. However, on the basis of a scientific experiment in [60] we can establish the fact that having a technology that is convenient to use and can easily interface with network to provide information to a computing processing, opens up the new potential for allowing remote patient exercise monitoring that is can be relied upon.

Proposed Software System Solution

The proposed software system is a layered architecture that consists of three layers: (1) mobile phone application, (2) a central hub that will act as a data store, (3) a doctor's web application that doctors can use.

The first layer, that is the mobile phone application, is basically an interface to communicate with the health bands/devices. All of the consumer grade health fitness devices communicate using the Bluetooth Low Energy communication mechanism to a smartphone. The type of application proposed in this layer is an application that can establish a Bluetooth connection with the device, or subscribe to the sensors of the device and continuously listen and monitor the sensor reading values. This application has been kept separate of any other layer and communicates with layer 2, i.e. the hub via REST API calls. The reason behind this is that REST API facilitates the process of layered architecture by allowing to be called from a large number of types of devices that are connected to the internet. The mobile application in layer 1 is defined to function in a way that it collects the data from the health sensors, such as heart rate, accelerometer data, etc. and pass it on to the hub using REST API as defined by the hub. This allows to account for a number of different kinds of mobile phone ecosystems, such as Android, iPhones and Windows Phones as well as the mobile phone applications can then again be processed with adding capabilities to connect with more than one type of health band devices. In this project, the mobile phone application is only programmed to communicate with one type of health band: Microsoft Band. However, additional devices may be added in future so the phone can pair to more devices for getting the required sensor readings and be capable of sending that data to the hub. A number of projects and applications already exist that exhibit this kind of feature where they can establish communication with more than one kind of health band and provide a single interface and ecosystem to deal with and share with your friends and family who may not have the same category of health devices. One example of such platform is Stridekick (previously known as Matchup.io) [45].

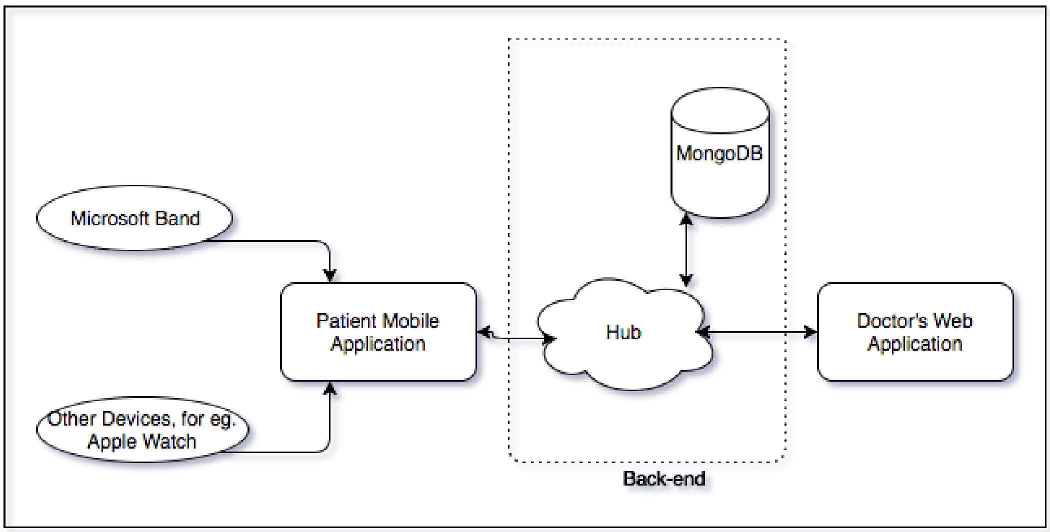

Figure 2-1 Architecture of the system

The second layer a central software layer that will act as a communication and data store hub for the application. This application will allow the mobile application to connect in a secure manner and authenticate itself to be able to transmit the sensor data for storage and further access by the doctor’s application. The hub, being a communications hub, is a REST API Server that is built on top of Node.JS backend JavaScript framework. It holds the infrastructure that is the backbone of this framework. This acts as a central layer of the framework which accepts the incoming data from patients about their activity sessions and from doctors about the goals for their patients.

All exchange of data takes place via the REST APIs that this hub provides. Model/Schema definitions have been identified in order for this kind of communication to take place. This is connected to a permanent data store that will be responsible to store all the data that is being recorded. This framework chooses a NoSQL database, MongoDB, as the backend storage for its persistent data storage needs because of its scalability and flexibility, instead of the regular SQL database. A comparison of NoSQL database MongoDB and a SQL database MySQL can be found at [46]. Having flexible software systems allow us to evolve projects in a much simpler manner avoiding setbacks and transition complexities while evolving the application incrementally. This feature has also been an integral role player and enabler in the evolution and development of this project by a rather small team. It also allows to focus on business logic [68] and use cases [69] first rather than spending time on boiler plate of other things that might come in the way. The framework that is backing our hub is called LoopBack [47]. It provides simple connectors in case one might change our minds to use a different kind of data store instead. However, it does not handle any data migration that might come into play at that moment and it will have to be handled manually or by using some other external data migration tool available. LoopBack also has easy commands to change the schema/models in a way where we can add more properties to the model or can add new models and new relations to the existing schema at any time we want without causing the whole application system to fail while the change is taking place. This layer will also have to take in consideration the securing of the data, the REST API endpoints and other concerns that relate to data privacy, and encryption of sensitive patient data. This addresses the concern by encrypting the personally identifiable patient data using a secure encryption mechanism, whereas securing all other communication on the REST API via HTTP’s Secure SSL protocol. All the REST API endpoints will also be accessible using only a valid client based token based authentication system that allows for interruption free transition to a different compute node when scaling the application up or down. The data model also allows to completely segregate the sensitive identifying patient data and to encrypt the values in the database itself. That also secures the data in the event that the database access is breached by some means. We also have the option to save the data in multiple data stores. That separation can be done on a model-to-model basis. One can choose to store particularly one schema/model on one database and other on a different one using our storage resources in much more flexible way.

The third layer is a client side JavaScript framework, Angular.JS web application. Angular.JS is a Model View Controller (MVC) or rather Model View Whatever (MVW) framework that was developed by Google on October 20, 2010 [48]. This layer provides the access to the physiologists using data from the hub. Doctors can use this web application to monitor the activity of their patients in an interactive manner. The activity session is visualized in a graph for the doctors to look at. Apart from that, it also has the capability of setting the exercise prescription, or in other words, the amount of exercise intensity the patient should be doing. Since this layer extracts and shows the doctors the sensitive patient data, security is a of great importance. We separate the sensitive data to being logically and schematically separate thus allowing us to use a deeper encryption mechanism where we are able to replace the patient data by encrypted blocks of strings. This web portal houses a number of other features such as the ability to add new patients, remove the patients from their list, allow doctors to change their login password. The ability to set a new weight or height recording is also available in this web portal. A responsive front end technology based on CSS (Cascading Style Sheets), Bootstrap has been used to develop the front end of this web application that will allow it to be accessed from different sizes of devices, like smartphones, tablets and computers with greater ease. Bootstrap allows to accommodate visual setting/configuration for multiple screen types by providing a toolset of CSS classes. It also has extra components that make use of JavaScript and jQuery, though we are not using any of those features explicitly in this application.

Architecture

An application that has a systematically defined architecture that separates the components based on their concerns and functionality is known to allow it to evolve in a more proper manner. This also allows a pluggable capability to the application in such a way that an independent third party can create an extension to complement the functionality of the existing application. The architecture has been defined into different components, separated by concerns and functionality. The logical first component is just the various kinds of health bands or devices that will be used to transmit the physiological vitals from the user's or patient's body. Even though these devices can vary immensely in the type of the device they are: ranging from being a chest strap to a smart watch or from a foot pod to a smart shirt embedded with sensors, they have one thing in common: Bluetooth Low Energy connection. So the connection arrow between the devices and the mobile app in Figure 2-1 is going to be the sensor reading values getting transmitted over a Bluetooth connection that will be initiated by the Mobile Application.

The application that runs on the mobile platform is kept independent of any other system such that it can perform its functions independently. However, it needs to communicate with the hub server to read the latest goal for the session, and at the end of the session to transmit the recorded session data to the hub for further processing and exchange. The mobile application is designed to be as streamlined as possible by not putting any extra functionality in it than is required. Much of the data processing, analytics and storage will be done in the hub. The function of mobile application is thought to be a substitute for an actual trainer to be present for tracking and monitoring activity levels and progress.

Internet connectivity is required for the mobile application to be able to transmit the information recorded and also to read the goal details from the server. This can either be achieved through the use of 802.11 Wi-Fi or a cell phone network internet connection such as 4G LTE. The transfer of data that is taking place here is secured by HTTP Secure SSL mechanism and is end-to-end encrypted. For any sensitive identifiable data of the patient, the protection is further increased a level by transmitting an encrypted block of text through the existing end-to-end encrypted secure HTTPS protocol. This is enabled by the REST APIs that the Hub is exposing in order to get data and send data. The mobile application is built on the Android, iPhone, Windows Phone platform which utilizes the libraries provided by the band manufacturers to communicate and access the sensor values. For example, Microsoft Band has a library that Microsoft provides called Microsoft Band SDK. This can be used within Android ecosystem to set up a connection and talk to the band in an abstracted format instead of a low level format. Microsoft provides similar libraries for iPhone and Windows Phone. Apple provides similar libraries for the iPhone to communicate with its devices such as Apple Watch and iPhone’s health features.

Technologies Used

The most popular, current and common technological stack is being used to create this application. With JavaScript being at the core of this system, it uses the MongoDB ExpressJS AngularJS NodeJS (MEAN) stack. As discussed in the architecture section, MongoDB is at the very back end of the software stack working to provide scalable and flexible data storing functionalities. The LoopBack API framework generates an application that basically runs on a NodeJS server on an ExpressJS Web Application Framework. On the front end, AngularJS is being used to provide maintainable and testable front end system. Description of each of the technologies used can be found in detail in the following text. That will be divided into basically two sections: back-end and front-end, also known as to be the server-side and client-side. The following figure describes the utility and application of a software technology stack:

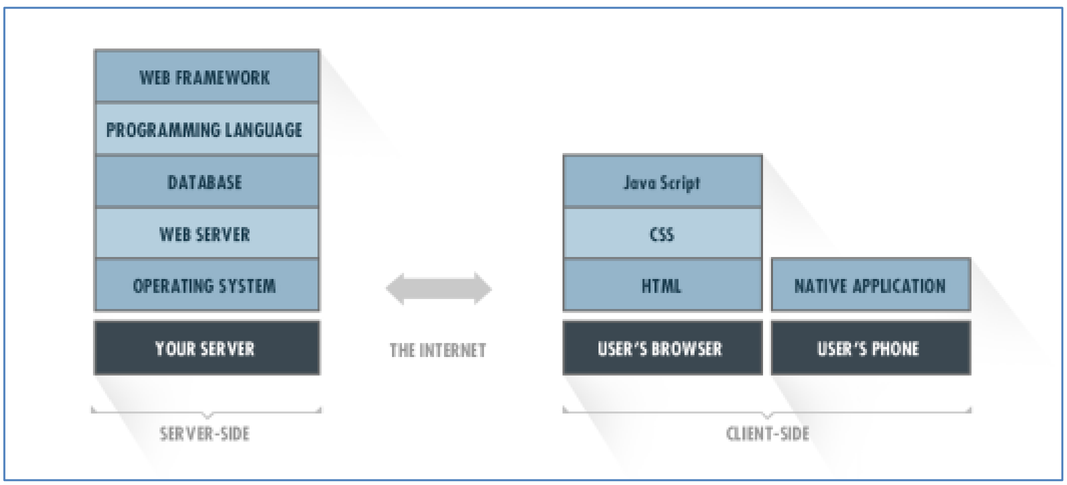

Figure 2-2 Tech Stack Setup Illustration (adopted from [49])

Using the illustration in Figure 2-2, one can easily demonstrate our technologies and where they belong. The Hub, as discussed in the architecture, is the central server. As shown in Figure 2-2, Web Server in this system is Node.js. Node.js is a platform that provides a runtime on the server side where JavaScript can be written on the server. JavaScript on the server side makes a server efficient by utilizing the non-blocking IO operations and an event loop instead of traditional threading models.

This Node.js webserver is a cross-platform runtime environment, which means it can run on any operating systems, such as Linux, Unix, MacOS, and Windows. This makes our application independent of the type of operating system we choose to run on. For deploying a Node.js application server, a virtualized or containerized Linux is a common practice. Hosting of our application server can be to achieved by using a dedicated machine. This benefits by having no surplus effort that may occur from any other type of configuration and any multi-node configurations. However, having a single machine as a back-end will easily become a bottleneck when the users of the application grow big and no longer the machine is capable enough to handle that load of processing. This bottleneck calls for the scaling of the back-end infrastructure from a single machine to something that can handle a bigger load of requests and processes. Historically it has been a practice to vertically scale the server (by increasing the performance of the server on which the application is hosted) when the need to scale arises. However, cloud computing technologies, specifically PaaS (Platform as a Service) allows us to scale horizontally by using multiple instances of the same type of virtualized server and easily scaling down when not needed. PaaS offerings usually take up a lot of configuration loads on themselves and the time and effort is saved when configuring and deploying the application server. Scaling down is just not as easily possible when a system is scaled vertically. Hence, it is preferred to use a virtualized or containerized Linux system as a go to solution to host our application server.

As for the database, we use MongoDB for its flexibility and scalability as compared to the more widely used MySQL [46]. MongoDB does not have a lot of functionalities that SQL Databases usually have. Some of these are: Auto-increment, Triggers and Stored Procedures. These are not included built in, but there exist work arounds by using external libraries. Although in the case of auto-increment, it is not advised to use auto-increment, because it will very easily become the bottleneck of a horizontally scaled up application where all new data must go through a central module. Now, we can decentralize the creation of auto-generated numbers but then again it becomes a huge computational workload of making sure generated numbers do not overlap.

As shown in Figure 2-2, the programming language is JavaScript. As discussed earlier, NodeJS allows JavaScript to be written even on the server side.

As in Figure 2-2, Web Framework would be ExpressJS, the backend of the MEAN stack. It is considered as a de-facto standard for writing web applications on NodeJS server [50]. The LoopBack [51] framework is used to get much of the backend work automated. “LoopBack is a highly extensible, open source NodeJS API Framework built on top of ExpressJS that allows to create dynamic end-to-end REST APIs with little or no coding” [51]. A design goal of the system is to delegate all the presentation layer code (for example, HTML) away from the backend to the client side frameworks. Whatever data transfer needs to be taking place will be through the REST APIs that the front end will manage to present in anyway it wants. That brings up to the Front-End and Presentation part of the system.

AngularJS is the front-end part of the MEAN stack that we are using for creating this application. AngularJS is a completely JavaScript based library that allows building MVC (Model-View-Controller) or MVVM (Model-View-View Model) or MVW (Model-View-Whatever) architecture for the client side data that is received from the REST APIs from the hub in our case. Its components complement Apache Cordova, which is a framework for cross platform mobile application development [48]. For the Doctor’s web application, angular is the client side framework whereas for the patient’s mobile application, it is a native Android/iPhone/Windows environment, clarifying the user's browser and user's phone of the client-side technology in Figure 2-2.

To complement the front end framework, and handling the presentation of the data in a nice and fluid UI, we use Bootstrap CSS and JS library [52]. It's another front-end framework that is designed to handle the presentation of the web applications. One of the big advantages of using Bootstrap is the capability of Responsive Web Design [53]. By Responsive design, what we really mean is that the web application view will adapt or rather, respond to the screen size they are viewed from. For example, a side navigation bar may be converted into a pop up menu on press of a button on the side instead of taking up the real estate and allowing to display the main content on the limited screen space instead.

Data Privacy and Encryption

According to [55], the privacy of a human being is of prime concern when handling and storing sensitive patient data because it facilitates fundamental values, including ideals of personhood such as Personal Autonomy, Individuality, Respect, and Dignity & worth as human beings.

When U.S. President Barack Obama stated in a speech regarding his economic recovery plan in 2009, that required necessary investments will be made to ensure that all of America's medical records are computerized in the next five years, [56] it raised immediate concerns regarding the privacy of sensitive patient data. For reasons discussed earlier in this section, it is very important to keep the data of patients secure and confidential, while at the same time ensuring the availability and accuracy of the data when stored digitally.

In regards to the proposal of data encryption and patient privacy related to the proposed software framework, this work proposes a system that combines the surety of privacy and the feasibility of the system at the same time.

The identifiable patient data should be completely separated from the other data that we are collecting. Identifiable patient data is considered the data that can reveal the identity of the patient whose data is being used, or the identity data about the patient itself that is considered sensitive. Possibility of unauthorized use of such data can be an example of data being not held securely. Information about patient's name, address, telephone number, or any similar detail has been considered as identifiable personal information. Unidentifiable data such as the heart rate session data – that consists of the timestamp and the corresponding heart rate reading with a reference to what patient is considered to be non-identifiable data because given access to this data, it is difficult to find what exact patient is being referenced.

Since the sensitive and identifiable patient data has been separated from the rest of the data that is being collected and held, all of that data is to be encrypted using a symmetric key encryption using a salt. Salting is a technique to add extra security when storing passwords or keys. A salt is random data that is added to the existing key or password to make the hashes of two similar password or keys different. This is usually done to defend against dictionary attacks and pre-computed rainbow table attacks [70]. The storage of the decryption key to this symmetric key will be done by asymmetric encryption, specifically by encrypting the symmetric decryption key of that patient, which is unique to that patient, by the patient's public key. The properties of asymmetric encryption ensure that this will be now be secure and be only unencrypted by the private key that is securely held by the patient. When talking about securely encrypting the sensitive identifiable data, it is meant that when this is stored in the database, it is converted into blobs of encrypted text that in themselves have no meaning. The process of extracting meaning out of it will be as follows: the symmetric decryption key that was used to encrypt this data will be needed to decrypt this data. Since this key is encrypted by patient's public key, and it is only available with the patient, patient will be the only person who is capable of decrypting the information.

The next consideration is about allowing the access to data to authorized personnel like doctors, or friends and family. This can be achieved by adopting the method used in the OpenPGP which is a format that is used in sending of secure emails [57]. After setting up the data in the system, the patient can authorize doctor, or other entity to access the data by accessing the key to the protected information and encrypting it with the public key of the doctor and then storing it. This will allow only the specified doctor to be able to decrypt the sensitive data and read it.

The model also accounts for recovery of possibly misplacing the key by patient or the doctor. If the private key of patient has been misplaced, is no longer accessible or compromised, an immediate lockdown on that record can be put by the system followed by a prompt to either a recovery person or a previously authorized doctor to start the recovery process for the locked down data of that patient. The procedure will be as follows: The system will guide the authorized recovery person or doctor through the steps to use his private key to allow the system to run a function that will decrypt the patient data and will re-encrypt using a newly generated key this time. This key will now be stored by encrypting with the doctor's public key again and will effectively terminate the access of the compromised private key for access to the confidential data. After this step is done, the system will delete the public key of the compromised user from the list of references to this data. Then, for the rest of the entries, the system's routine will re-encrypt the key using their public keys. This step will ensure that none of the other users lose access to the data they are entitled to. However, it is important that this routine or function can be only run by one of the authorized user when at least one of the other authorized have authenticated themselves using different means that may possibly include a biometric or secure key generator or email, phone number and other password or security question – answers type mechanism and thus authorizing the system to allow the doctor to reset the keys. This is necessary to ensure that not any user authorized to have access to that data can just go and possibly revoke access to the patient data from everyone else who has access to it.

Once the key to the patient data has been reset, the patient will then be allowed to run another system routine or function that will now setup the new public and private key pair for the patient. This will enable patient to regain the access to his data with minimal risk of potential harm and breach of confidentiality.

However, apart from having this system in place, we propose allowing some extra protection features to be included in the system that will account for further reducing the risk of potential breach by identifying abnormal patterns or anomalies and adding a secure multi token based authentication when access is made in a suspicious way. This will minimize the risk that exists between the moment that the user loses his private key and the moment he realizes and takes action to secure his private data. Anomalous patterns can include things such as a change of device which the key is being provided from, geographical location of the IP address the access is requested from, attempt to read or access data that is not frequently accessed by the user or has not been accessed by the user. There should also be a system level timeout for sharing the data that if the person who the data is shared with does not access it for more than a specified interval of time, the access will automatically be suspended and will have to get re-authorized by the owner of the data.

With having this system in place, all the other data transmission and communication is also done in a secure, encrypted manner. The standard HTTPS secure SSL/TLS protocol is mandatory to be used for all communication within the application.

Implementation

The implementation of this project has been done to allow agile techniques of software development where the traditional work flow is replaced by an incremental and feedback driven development with much more involvement from the clients who set the requirement. In regards of this thesis project we have been glad to have the committee members and Dr. Zhang and Mr. Turner from the Presbyterian Hospital, Dallas to get involved in the feedback and incremental development of the software project. The deployment of the software framework has been done in a micro-service based architecture. Each of the different components have been instead of setting up software stack on one machine or multiple computer systems, has been used as a service. This holds for database, the server platform, and the web hosting space.

The Hub

The whole application is built and structured around a central API Server, called the Hub. The language for this API Server is JavaScript running on a NodeJS platform. Since this is the most important element of the application, and is also a centralized service, it provides two main concerns: scaling bottleneck and denial of service because this could easily be the single point of failure. Choosing an efficiently scalable platform as a service is thus the most suitable solution for the execution of this module.

One of the logical and simple system of deploying this system could be easily on one virtualized machine on an Infrastructure as a Service for example, an Amazon EC2

Linux box. However, this is not preferred because a runtime or a platform as a service will be better able to monitor and scale as compared to the machine. Also, runtime can be managed in a much more cost and storage efficient manner. However, this also means that we the data storage should now be handled independent of the running application because these services don't provide provisions for persistent disk storage. However, the LoopBack tool eases this for us by easily routing all the data to a MongoDB database without the effort of managing it. LoopBack is also a great tool to automate the entire backend development process to giving input your data schema to the LoopBack generator application and then configuring REST APIs to the models. The accesses to these APIs can be easily maintained with the use of Access Control List (ACLs). IBM Bluemix Cloud Foundry has been chosen to deploy this application because of its cost effectiveness and reliability. However, the provider of this service can be easily switched to any other provider with small effort of configuration. The scalability settings have not been configured for IBM Bluemix yet and the application only runs in a single node container of the Cloud Foundry which is accessible to the public world which is how other parts of the application will be able to access it. However, the APIs will be closed only to authenticated access, the system still needs to have a public facing access point.

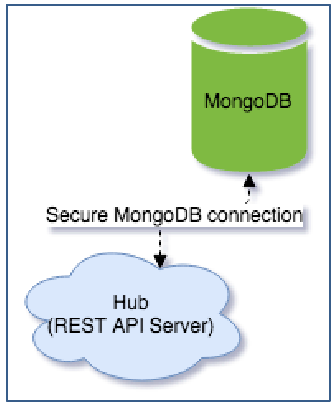

Figure 2-3 Data Flow: Hub

Even though the database schema can be developed in a highly incremental manner in this model design, we came up with an initial model such that we can provide a minimal viable product which can serve as a proof of concept as well as an acceptance test of the final working solution of the software framework. The database schema was thought upon and has incrementally evolved to as depicted in Figure 2-4.

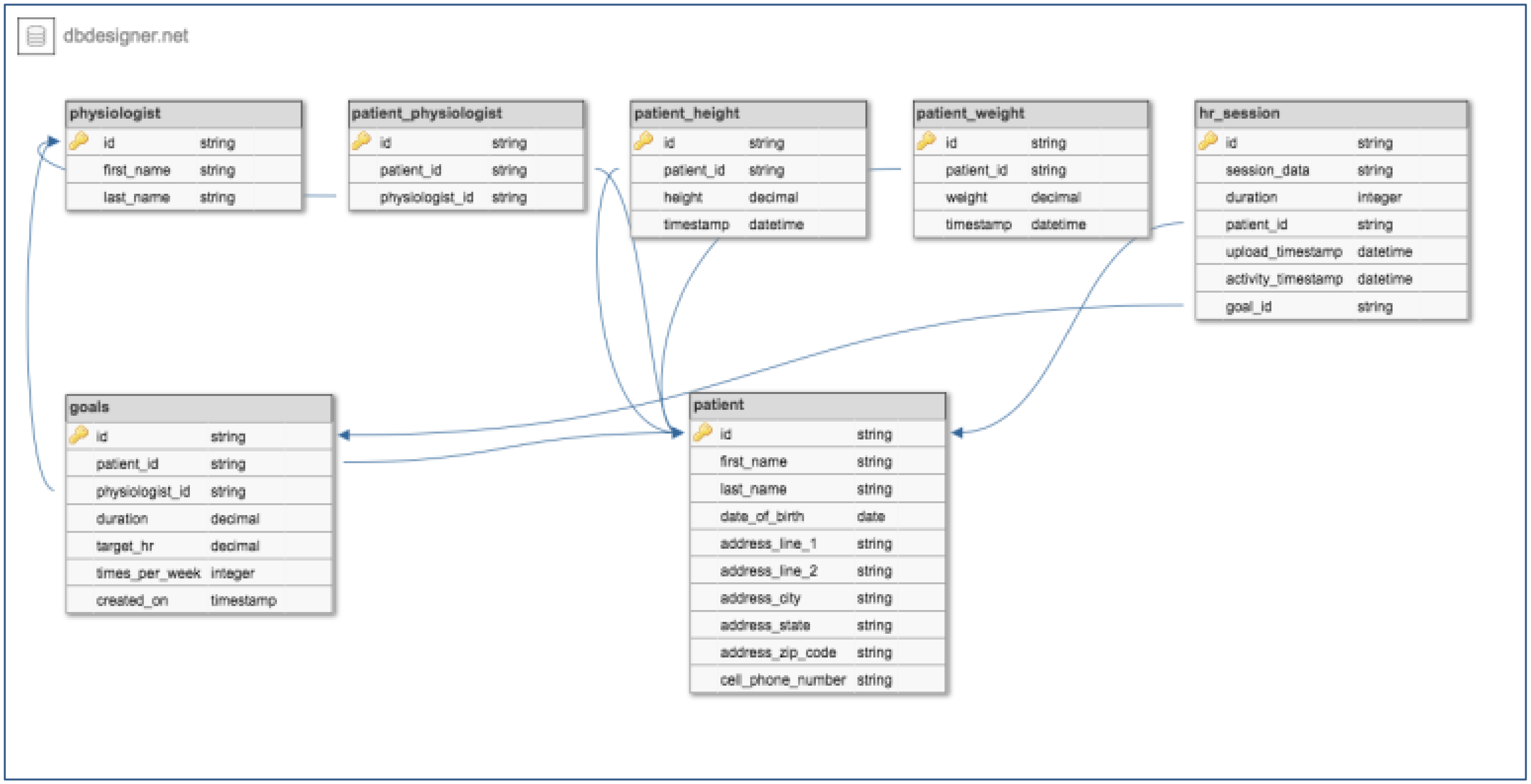

Figure 2-4 Database Schema

The MongoDB hosting, also is publicly available, one of the providers of MongoDB Hosting, MongoLab, now known as mLab has been chosen as the hosting provider for the MongoDB Database.

Patient Mobile App

This is an Android application that connects to a Microsoft Band via a Bluetooth connection and subscribes to the heart rate sensor. The patient's goal is retrieved from the hub and displayed on the screen. It then allows the user to start recording the heart rate and accelerometer data. The sensors from which to collect data can be configured in the doctor's web application. The UI is designed to be minimalistic for the sake of simplicity and convenience. The application will now display a timer and will start recording the data. Once the user stops the exercise session, then the application pushes the recorded data over to the Hub using a secure REST API which then saves the data persistently in a MongoDB database.

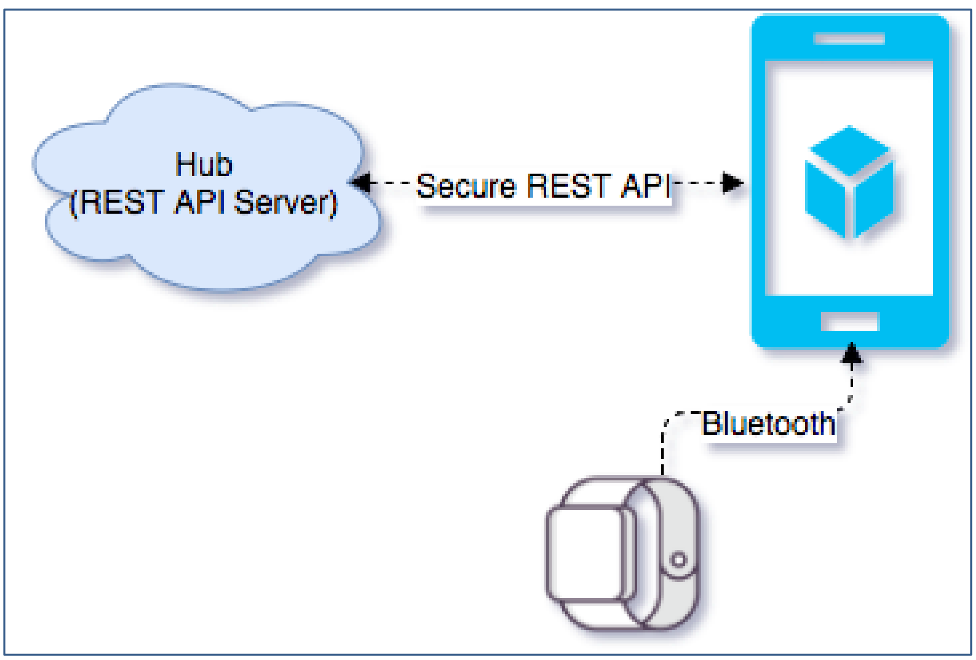

Figure 2-5 Data Flow: Patient Mobile App

Internal design of android application is divided into two main components: The Main Activity thread and a Background Thread that continuously listens for the new heart rate data. As soon as the application is started up, the sensors will be attempted to subscribe in the background. The recording of data only starts when such is indicated by the main activity thread and at that point a file “sessiondata.json” is opened and the data is written in it. Once the main activity indicates a Stop action, the thread stops recording the data and will now start uploading the data to the hub.

Physiologist Web Application

Google Firebase, a reliable and scalable hosting service that provides SSL support has been chosen to host the doctor’s side, static AngularJS web application. This can easily be any web hosting service and will be the most economical deployment because the scalability is not the biggest concern at this level as much as availability. The reason is, that once the web application has been requested from the web hosting, everything else runs directly on the client browser receiving data from the Hub. One of the many advantages of using client side JavaScript framework like Angular was that this app does not need any server side processing. It is completely isolated from the system and communicates with the hub directly from a client machine’s browser.

As depicted in Figure 2-6, the Doctor’s application is written in a client side JavaScript Framework, AngularJS. This allows us to remove the need of having a separate server hosting the web application. Instead, we can cleanly interface the client application running inside of the web browser on the client’s end to communicate with our Hub. This makes a clean design and is a less hassled evolution of the application.

AngularJS is a client side MVC model developed by Google. This allows the client side JavaScript code to be written in a testable and manageable way. The view changes are based right on the browser end, and the controller manages the state of execution and manages all the functionality of the view. The code can be efficiently used within Angular using Components. This also allows someone new who is picking up on the development of application to understand quickly by just looking at the client side code.

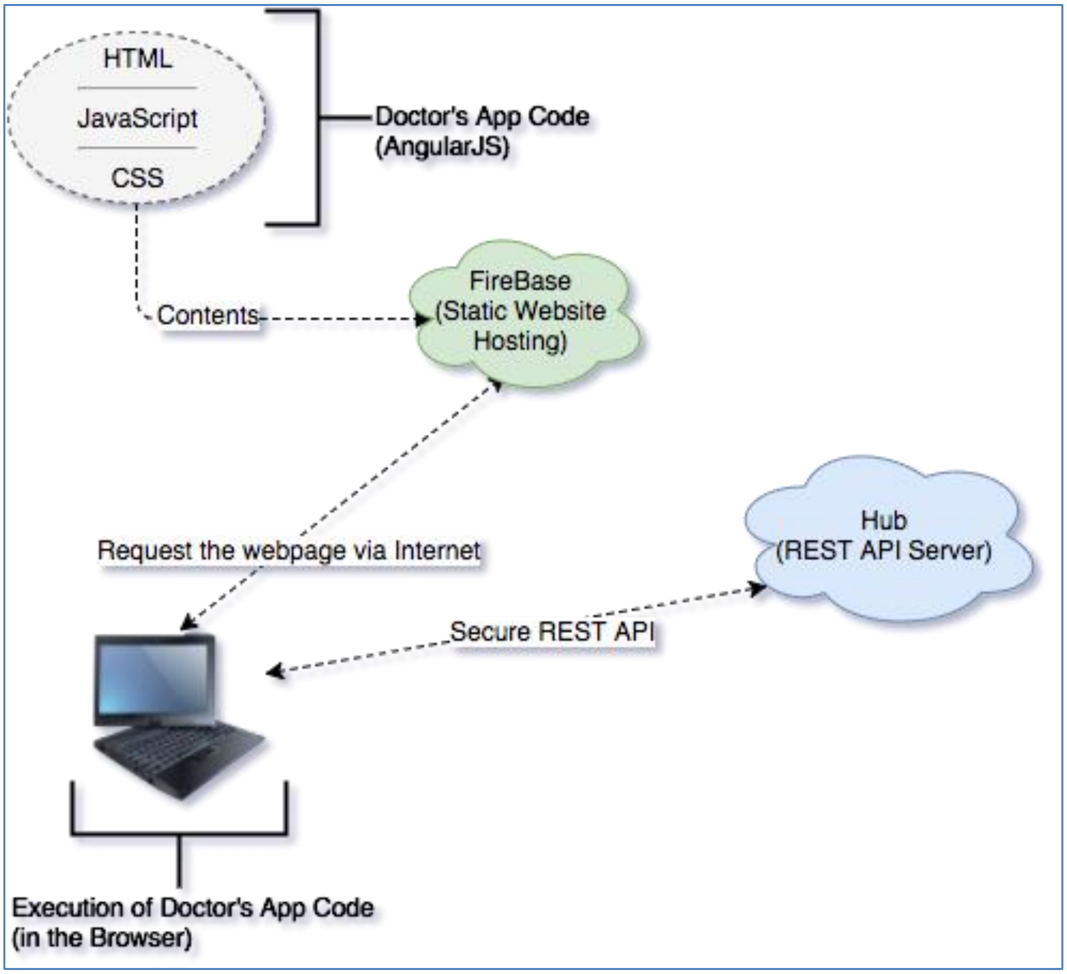

Figure 2-6 Implementation of Doctor's Application

A good practice in MVC Architectures is to not let controllers get bigger than one page, or in other words, if you need to scroll to see the whole controller either horizontally or vertically, then it has been written in a wrong way. This then allows for there to be much better layering within the client application using services. A service is a module within the application that is responsible for computing, retrieving and making the data ready. This is also a very good way to reuse any existing functionality across multiple controllers because the services are capable of providing a bigger scope that can be specified from being accessible to a whole module and/or page.

Angular allows Dependency Injections to be used just as they are used in Java and C# but have always been very uncommon in a JavaScript framework. Different functionalities are achieved by utilizing different services and modules for example to make http requests, you would use the $http [72] service. For client side routing, we have chosen the more versatile state based ui-router instead of the more commonly used ng-route module [71]. Authentication is manageable as a service and is a client token based system. This is better because at high scalability and multiple server hosts running, client telling who they are is much intuitive than maintaining a state in the server. This has usually been achieved by server side authentication but it doesn’t scale well. Having client side authentication goes will with receiving the next response from a completely different mirror of the same application.

Limitations

The patient data privacy and encryption mechanism used in this system is reliable but the case of users losing or getting their private keys compromised or locking themselves out in this manner poses a problem of effectively managing this solution without having a central authority or system to have access to it unless it is a government authority.

The feedback system from physiologists to the patient can currently be only done in one way. More experiment and feedback will be needed from physiologists and patients about what kind of feedback would be most effective. Knowing that, we can then provide more than one way to set goals.

This platform is only useful if the physiologists know about it and are willing to set the patient’s smartphones with the companion app. A more versatile flow would be allowing patient to store their data regardless of doctor and then allowing a physiologist to take a look at that data by sharing it with him. This allows for both ways of propagation: a patient can also now let doctor know about the existence of such a system for them that enables them to monitor exercises and activity levels remotely.

Background running of the patient mobile application is a current limitation where the application has to be in the foreground for it to continue recording your exercise or activity session even if the user wants to leave the application momentarily or for the duration of activity and then come back to stop the session. A better way of handling this would be using a service on the Android platform that will run your job in the background and will continue even when the app is not in foreground. Utilizing the notification area to provide a quicker way to stop the exercise session will be ideal.

The current platform does not allow the monitoring to be in a real time fashion. This can be made possible by sending each bit or reading of data to the hub and letting doctor see it in real time by opening a streaming connection to the Hub to listen for changes and updating the graph as soon a change is received by the Hub and transferred to the doctor’s web application. However, this cannot be done effectively just by using REST APIs. A streaming capability of some kind or client-server evented communication will be required to make this possible in the context of this application. However, this will require modifications because the structure of APIs and the database will have to be modified. Angular and Android part should comparatively require fewer modifications.

A richer platform for wearable devices, for example Apple Watch’s WatchOS, the need of a mobile application can be completely removed and the application can be made for running and doing everything on the wearable platform itself. However, none of the wearable devices other than Apple Watch and possibly a few Android Smartwatches allow apps to written in such a format.

Chapter 3

Result

Functional Overview

The overall system has been developed with usability, flexibility and scalability in the core of the design and development process. The patient mobile app is a simple app that makes bluetooth connection to the health band on patient's body and transmits sensor data to the hub, which is a central server in this system. The hub is also equipped with secure REST API endpoints that allow seamless communication between various functional parts of the system. The patient mobile application uses REST APIs to transmit datat to the hub and store recorded health data on the MongoDB backend. Another key feature of patient's application is the fetching of goal that has been set by the doctor for the particular exercise session. The hub server employs standard encryption practises to handle the privacy and confidentiality of sensitive patient data while maintaining all the transmission paths to be secured using HTTP SSL protocol. The doctors web application is an extensible front end built using Angular.js and Bootstrap. While angular allows separation of client side presentation layer code from the server side by utilizing the REST API endpoints to get data, Bootstrap helps give the website a lucid and responsive look and feel across multiple sized devices. The patient's web application allows for various basic functions with specific implementations for key functionalities such as sensor selection, goals, and data viewing. The doctor can choose from available sensor values and options, which one to view at a particular instant of time. This has further been illustrated in the upcoming section where User Interfaces are displayed and discussed.

User Interfaces (UI)

The user interfaces will be divided into two major sections: a patient side UI and a doctor side UI.

The patient side UI has been designed with an idea in mind: they don't need to know or do anything more than what is entirely necessary. Hence, the patient's mobile app has a straightforward indication of the current exercise intensity goal and only a simple indication of heart rate and its quality, with a start exercise button.

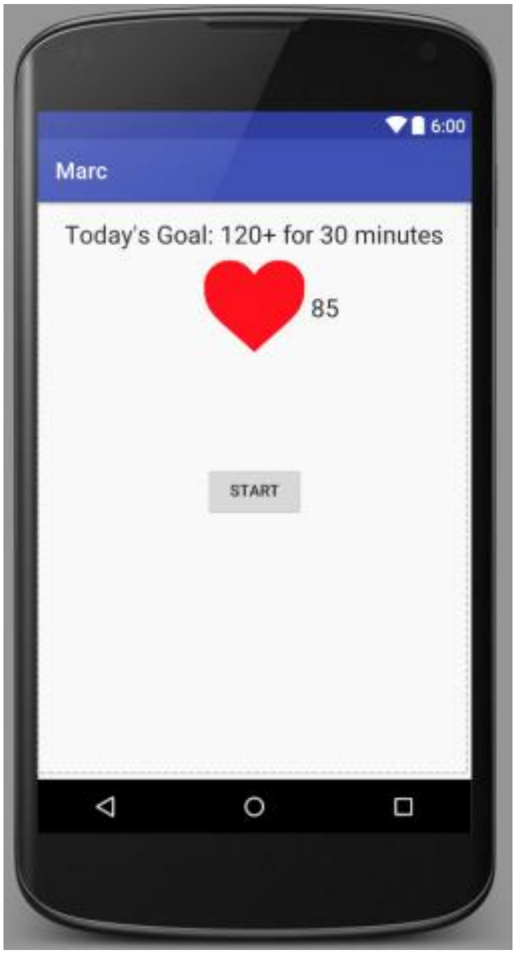

Figure 3-1 Screenshot: Patient Mobile Application

Once the user is ready to start the exercise session, and the Start button is pressed, the next screen is displayed in Figure 3-2. The goal provides a reference or a prescription of intensity level that is for the user for the duration of the exercise.

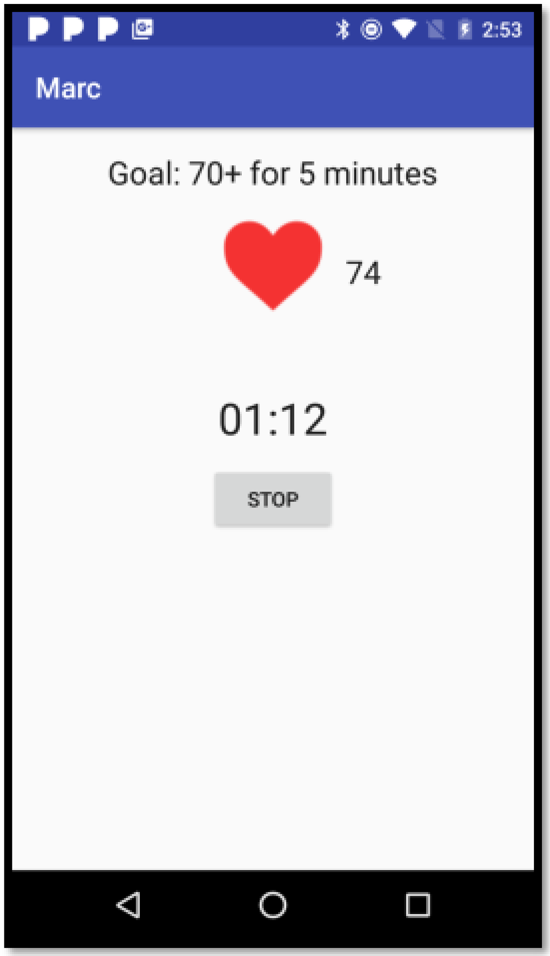

Figure 3-2 Screenshot: Ongoing Exercise Session Screen

For all other kinds of infographic and activity during the exercise workout, the patient can easily make use of the existing band or device's companion app that typically provides a lot of metrics such as the calorie burn, number of steps, miles and a graph of heart rate performance.

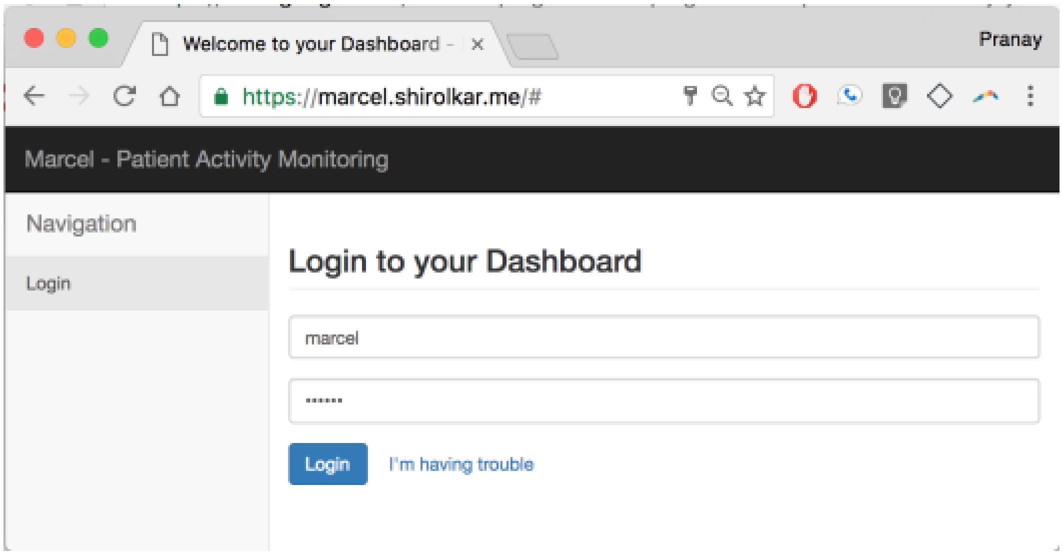

Figure 3-3 Screenshot: Doctor UI Login Screen

The doctor's side of the application is a web application that is protected by a username password combination and the UI looks like as depicted in the Figure 3-3. A web application is chosen so that the representation of all the details can be done in a way where it can be represented nicely on a number of devices without any installation. A doctor can thus easily switch between his computer, tablet or a cell phone to be able to use the software application.

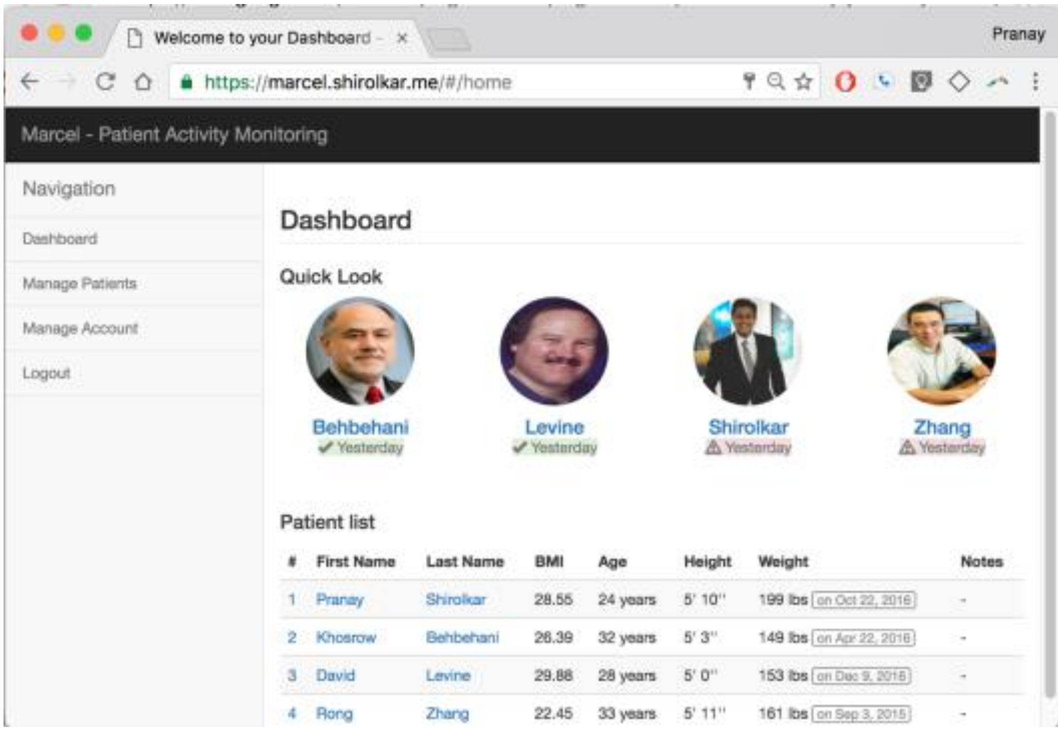

Figure 3-4 Screenshot: Doctor UI Dashboard

After the login, dashboard view is presented as illustrated in Figure 3-4. The dashboard has a “Quick Look” which contains patients who are frequently accessed and the bottom section is a full section of the available patients for the logged in doctor. The markers below the names on the dashboard’s quick look displays a color coded indication of the exercise status for that particular patient.

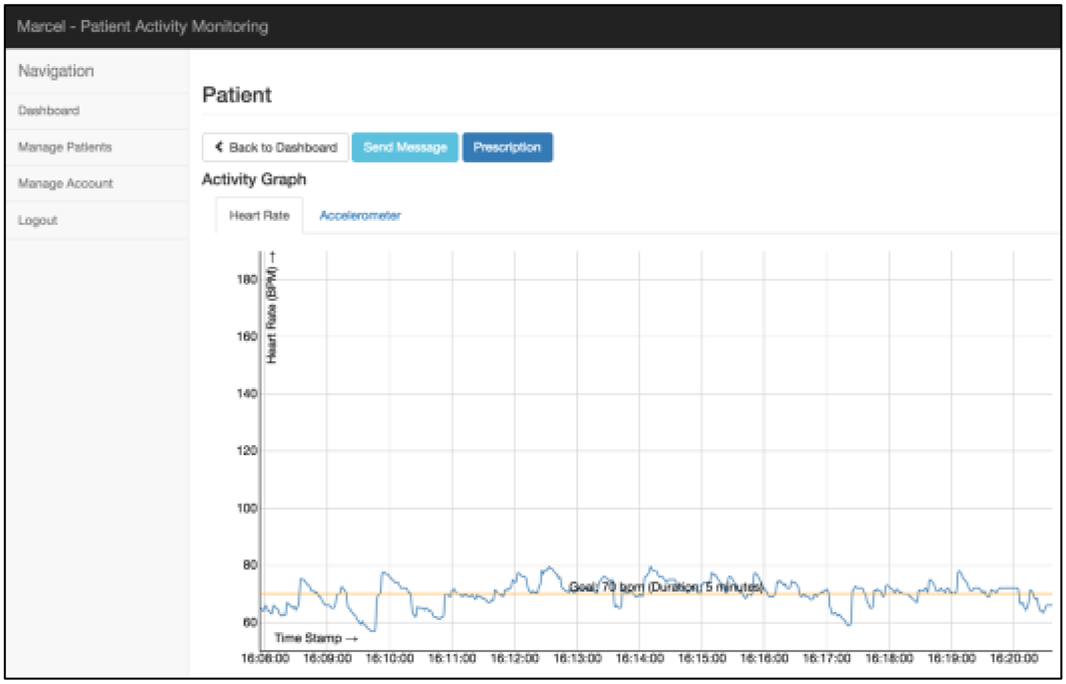

Figure 3-5 Screenshot: Doctor's UI

A central feature of doctor's User Interface is to visually demonstrate the data that has been collected from the patient's side of the application. This has been illustrated in Figure 3-5. This can be best done using ECG-style graphs overlaid with different parameters such as the goals that this work proposes.

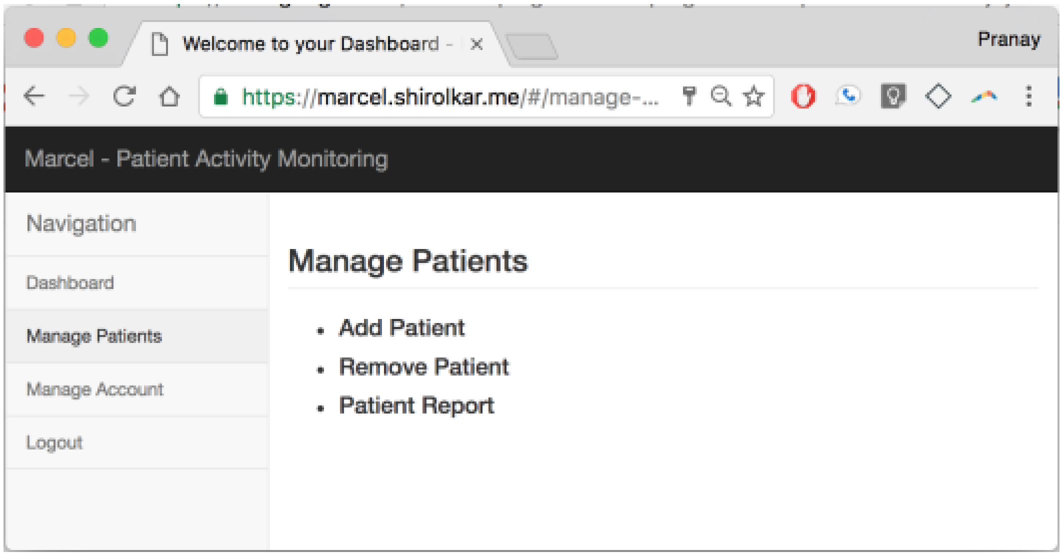

Figure 3-6 Screenshot: Doctor UI Manage Patients Page

Other functionality is also important which allow the doctor to easily manage his account, including changing his account and login credentials, manage patient list, etc.

Chapter 4

Discussion

The wearable technology is now capable of better integrating with one's lifestyle and has motives of increasing the positive impact on the users using new technology. The wearable fitness devices that range from being a wrist band, chest strap to a shirt that is embedded with sensors and other standalone devices such as foot and belt pods are changing the way people exercise and are motivating more people to exercise by providing useful information, activity based goals and alerts and a way to get others motivated by sharing the accomplishments with them. The technology has been endorsed with scientific studies for its accuracy and a lot of software applications are now increasingly integrating with it.

This technology with the ability to allow doctors to intervene and get access to this exercise data and for patients or users who need personalized care, this thesis concludes that a software framework that allows necessary platform for users to share their data with doctors as well as for doctors for a way to enable this kind of remote feedback. Current technology available for this work is either too clinic-facility oriented or is not available for an affordable price.

A software platform will be capable of allowing this kind of communication and will transform the way doctor's currently get feedback about their patient's activities and exercises without investing substantial capital in getting access to it.

Mr. Turner, who is an exercise physiologist and works with patients who exercise based on a prescription at home, has provided feedback in regards to the web application that has been developed as being "great and relatively easy" to navigate. A software system to manage everyday health metric data will be useful to be integrated as an augmented source of medical and physical history of data of the patient for the doctors. The added capability of setting prescriptions and goals within the application's ecosystem provides for a better management of remote exercises.

Chapter 5

Future Work

The platform suggested in this thesis can be extended towards availability of the mobile app for general public. The system would then be modified where it is substantially equipped with data analytics, data mining and machine learning capabilities. These capabilities will then allow the system to work autonomously towards gathering data from a large mass and providing customized feedback to users on their health and fitness goals based on their Body Mass Index (BMI), ethnic group and age group. This will dramatically allow the system to expand and increase its impact on the society. Also, the platform logic could notify the users of the anomaly that it found and that it could be a potential symptom of an illness which could be taken care of by a professional physician. Having the historical access to health metrics will then provide a great deal of insight when the doctor examines the patient. The existing data and the feedback system of the system can then be coupled with the doctor’s end of the system, facilitating the constant monitoring of health data and allowing doctor to set custom fitness goals for that particular patient.

The mobile phone application can be made more versatile and capable of accommodating more than one type of health device, applications such as Matchup [45] will allow the users to choose one of the many options for health bands available or choose one they like, or is more economical, more durable, has more battery or has better accuracy, or something that will merge better in their existing line of devices, for example someone might prefer to buy an Apple watch if they already have an Apple iPhone or someone with Android phone may prefer an Android Smartwatch or with Windows. There can be other ways of developing this mobile application. One way is to develop one application to communicate with one type of band and have people use the appropriate one when they need it. One other possibility is to incorporate all the types of devices in one app itself and let the user decide which one they are pairing the application with. However, this may result in the application code becoming unnecessarily bulky and not appropriate for people who don't have a lot of storage capacity on their mobile phones or for people who have older phones. A solution to this problem may be to dynamically be able to import packages or libraries to talk to a particular type of device after the user has installed the app. On the first start-up of the app, when asking for the type of device they are pairing with, the application should be able to request and install the appropriate packages or libraries required to communicate with that particular device. This way it will also be easier to add new devices over time and keep all of them easily maintainable.

The monitoring of patients can also be made partially real-time by changing the way patient data is transmitted through the app to the hub. If the app is configured to send the data at every short interval of time, ideally at the very moment a reading is obtained (a typical frequency for readings is 1 Hz, or 1 reading per second). This can allow us to display status to the doctor's end that the patient is doing the exercise session at the present time and can be provided with a real time graph of his exercise.

Cross platform software development framework like Ionic can be used to create the doctor's side of the web application. Frameworks allow us to write the code once, using HTML5 and CSS and JavaScript, and it will then be available for use as an app on the Web, Android, iPhone, etc. One can't necessarily do cross platform development of the patient's app because of its necessity to utilize many of the native features and libraries in order to communicate with the health bands.

References

[1] Davies, Peter, and Gerard Garbutt. “The Exercise Prescription.” The British Journal of General Practice 60.577 (2010): 555–556. PMC. Web. 8 Oct. 2016.

[2] Chief Medical Officer, Department of Health, UK. “ At least five a week: Evidence on the impact of physical activity and its relationship to health.” http://webarchive.nationalarchives.gov.uk/20130107105354/http://www.dh.gov.uk/en/Publicationsandstatistics/Publications/PublicationsPolicyAndGuidance/DH_4080994 retrieved on 9 Oct. 2016.

[3] Apple, Inc. “Introducing ResearchKit.” http://researchkit.org retrieved on 10 Oct. 2016.

[4] “Accelerometer” Wikipedia. Web. https://en.wikipedia.org/wiki/Accelerometer retrieved on 10 Oct. 2016.

[5] Bildr.org. “Sensing Orientation With The ADXL335 + Arduino” http://bildr.org/2011/04/sensing-orientation-with-the-adxl335-arduino/ retrieved on 10 Oct. 2016.

[6] Microsoft, Inc. “Microsoft Band sensors” https://support.microsoft.com/en-us/help/4000323/band-hardware-sensors retrieved on 10 Oct. 2016.

[7] Ryan Goodrich. “Accelerometer vs. Gyroscope: What's the Difference?” http://www.livescience.com/40103-accelerometer-vs-gyroscope.html retrieved on 10 Oct. 2016.

[8] “Photoplethysmogram” Wikipedia. Web. https://en.wikipedia.org/wiki/Photoplethysmogram retrieved on 10 Oct. 2016.

[9] David Nield. “How it works: We explain how your fitness tracker measures your daily steps” http://www.wareable.com/fitness-trackers/how-your-fitness-tracker-works-1449 retrieved on 10 Oct. 2016.

[10] Empatica, Inc. “E4 wristband” https://www.empatica.com/e4-wristband retrieved on 10 Oct. 2016.

[11] JW Bos. “UV Optical sensing” http://www.wikid.eu/index.php/UV_Optical_sensing retrieved on 10 Oct. 2016.

[12] “Microsoft Band” Wikipedia. Web. https://en.wikipedia.org/wiki/Microsoft_Band retrieved on 13 Oct. 2016.

[13] Matt Fitzgerald (18 June 2005). Runner's World The Cutting-Edge Runner: How to Use the Latest Science and Technology to Run Longer, Stronger, and Faster. Rodale. pp. 63–. ISBN 978-1-59486-091-1. The Finnish company Polar Electro, founded in 1977, was the first to sell wireless heart rate monitors for athletes and still dominates this market.

[14] “Apple Watch” Wikipedia. Web. https://en.wikipedia.org/wiki/Apple_Watch on 20 Oct. 2016

[15] Carly Page. “Apple Watch is 'world's best selling wearable' with 4.2 million shifted in Q2” http://www.theinquirer.net/inquirer/news/2418517/apple-watch-is-worlds-best-selling-wearable-with-42-million-shifted-in-q2 on 20 Oct. 2016

[16] Jill Duffy. “Polar A360” http://www.pcmag.com/review/343936/polar-a360 on 20 Oct. 2016

[17] James Stables. “Polar A360: Polar's new HR tracking fitness band explained” https://www.wareable.com/polar/polar-a360-features-release-date-specs-price on 20 Oct. 2016

[18] Paul Lamkin. "Polar M600 review" https://www.wareable.com/android-wear/polar-m600-review on 20 Oct. 2016

[19] Timothy Torres. "Fitbit Charge 2" http://www.pcmag.com/review/347317/fitbit-charge-2 on 20 Oct. 2016

[20] Francois Xavier Gamelin, Serge Berthoin and Laurent Bosquet. "Validity of the Polar S810 Heart Rate Monitor to Measure R–R Intervals at Rest" Medical Science Sports Exercise, Volume 38, No. 5, pp. 887-893, 2006. http://sportexperts.org/publication/17.pdf on 20 Oct. 2016

[21] Fitdigits. "The Importance of Resting Heart Rate" http://www.fitdigits.com/resting-heart-rate.html on 20 Oct. 2016

[22] Mayo Clinic Staff. "Exercise intensity: How to measure it" http://www.mayoclinic.org/healthy-lifestyle/fitness/in-depth/exercise-intensity/art-20046887 on 20 Oct. 2016

[23] Melanie Greenwood. "How Important Is Heart Rate Monitoring During Exercise?" http://www.livestrong.com/article/374396-how-important-is-heart-rate-monitoring-during-exercise/ on 20 Oct. 2016

[24] Morris, J.N. & Hardman, A.E. Sports Med (1997) 23: 306. doi:10.2165/00007256-199723050-00004

[25] Gottlieb DJ, Redline S, Nieto FJ, et al. Association of usual sleep duration with hypertension: the Sleep Heart Health Study. Sleep. 2006;29(8):1009-14. https://www.researchgate.net/profile/Carol_Baldwin/publication/6843573_Association_of_usual_sleep_duration_with_hypertension_The_Sleep_Heart_Health_Study/links/00b7d522415afd7f74000000.pdf 20 Oct. 2016

[26] "Subjective Sleep Characteristics of 1,485 Males and Females Aged 50–93: Effects of Sex and Age, and Factors Related to Self-Evaluated Quality of Sleep"

J Gerontol A Biol Sci Med Sci(1996) 51A (3): M108-M115. doi: 10.1093/gerona/51A.3.M108 http://biomedgerontology.oxfordjournals.org/content/51A/3/M108.short on 20 Oct. 2016

[27] Scott AJ, Ladou J. Shiftwork: effects on sleep and health with recommendations for medical surveillance and screening. Occup Med. 1990;5(2):273-99. http://europepmc.org/abstract/med/2203158 20 Oct. 2016

[28] Thomas Kohnstamm. “Microsoft band, the first wearable powered by Microsoft Health, keeps fitness and productivity insights a glance away” https://news.microsoft.com/features/microsofts-new-cloud-powered-wearable-keeps-fitness-and-productivity-insights-a-glance-away/#sm.0001iytz363glffpw0l1z3w007epo on 20 Oct. 2016

[29] Health Research Institute, PWC. “Health wearables: early days” https://www.pwc.com/us/en/health-industries/top-health-industry-issues/assets/pwc-hri-wearable-devices.pdf on 20 Oct. 2016

[30] Margaret Rouse. “REST (representational state transfer)” http://searchsoa.techtarget.com/definition/REST

[31] Omsignal. “OMsignal Launches New Smart Sports Bra; Offers Truly Feminine Wearable Tech That Goes Beyond the Smartwatch and Wrist-Based Trackers” http://www.prnewswire.com/news-releases/omsignal-launches-new-smart-sports-bra-offers-truly-feminine-wearable-tech-that-goes-beyond-the-smartwatch-and-wrist-based-trackers-593948571.html on 20 Oct. 2016

[32] Jess Bolluyt. “Here’s What You Can Really Do With Apple’s Health App” http://www.cheatsheet.com/technology/the-best-new-apps-that-use-apples-healthkit.html/?a=viewall on 20 Oct. 2016

[33] Epic Systems. “MyChart” https://itunes.apple.com/us/app/mychart/id382952264 on 20 Oct. 2016

[34] Chris Comish. “ UT Southwestern led study shows heavy use of MyChart technology platform among cancer patients” https://bionews-tx.com/news/2014/07/11/ut-southwestern-led-study-shows-heavy-use-of-mychart-technology-platform-among-cancer-patients/ on 21 Oct. 2016

[35] Adam Clark Estes. “Apple's HealthKit Now Sends Medical Data Right to Your Health Record” http://gizmodo.com/ihealths-healthkit-integration-sends-data-right-to-your-1635808368 on 21 Oct. 2016

[36] Google, Inc. “Rest API” https://developers.google.com/fit/rest/ on 21 Oct. 2016

[37] Jonah Comstock. “App developers get first chance to work with Google Fit” http://www.mobihealthnews.com/35674/app-developers-get-first-chance-to-work-with-google-fit on 21 Oct. 2016

[38] Medtronic. “ZephyrLIFE™ Home Remote Patient Monitoring” http://www.medtronic.com/covidien/products/health-informatics-and-monitoring/zephyr-life-home-remote-patient-monitoring on 21 Oct. 2016

[39] Medtronic. “Remote Patient Monitoring” http://www.medtronic.com/innovation-au/remote-patient-monitoring.html on 21 Oct. 2016

[40] Medtronic. “Comprehensive Remote Monitoring With MyCareLink Patient Monitor” http://www.medtronic.com/us-en/patients/treatments-therapies/pacemaker/living-with/carelink-comprehensive-remote-monitoring.html on 21 Oct. 2016

[41] Medtronic. “Medtronic CareLink Network for Cardiac Device Patients” http://www.medtronic.com/us-en/healthcare-professionals/products/cardiac-

rhythm/patient-management-carelink/medtronic-carelink-network-cardiac-device-patients.html on 21 Oct. 2016

[42] Ogunduyile O.O ., Zuva K ., Randle O.A ., Zuva T. " UBIQUITOUS HEALTHCARE MONITORING SYSTEM USING INTEGRATED TRIAXIAL ACCELEROMETER, SPO2 AND LOCATION SENSORS." International Journal of UbiComp (IJU), Vol.4, No.2, April 2013. Web. 21 Oct. 2016

[43] "Body Area Network" Wikipedia. Web. https://en.wikipedia.org/wiki/Body_area_network on 21 Oct. 2016

[44] "IEEE 802.15.6" Wikipedia. Web. https://en.wikipedia.org/wiki/IEEE_802.15.6 on 21 Oct. 2016

[45] Brad Merrill. "Matchup Launches Cross-Device Fitness Tracker App For iOS" http://venturebreak.com/2015/06/12/matchup/ on 23 Oct. 2016

[46] MongoDB. "MongoDB and MySQL Compared" https://www.mongodb.com/compare/mongodb-mysql on 23 Oct. 2016

[47] LoopBack. "The Node.js API Framework" https://loopback.io/ on 23 Oct. 2016

[48] "Angular JS" Wikipedia. Web. https://en.wikipedia.org/wiki/AngularJS on 23 Oct. 2016

[49] Gil Edelman. "How to choose your tech stack" http://svsg.co/how-to-choose-your-tech-stack/ on 24 Oct. 2016

[50] Paul Serby. "Case study: How & why to build a consumer app with Node.js" http://venturebeat.com/2012/01/07/building-consumer-apps-with-node/ on 24 Oct. 2016

[51] IBM / StrongLoop. "The LoopBack framework" https://docs.strongloop.com/display/public/LB/LoopBack on 24 Oct. 2016

[52] "Bootstrap (front-end framework)" Wikipedia. Web. https://en.wikipedia.org/wiki/Bootstrap_(front-end_framework) on 24 Oct. 2016

[53] "Responsive Web Design" Wikipedia. Web. https://en.wikipedia.org/wiki/Responsive_web_design on 24 Oct. 2016

[54] Benaloh, Josh, Melissa Chase, Eric Horvitz, and Kristin Lauter. "Patient controlled encryption: ensuring privacy of electronic medical records." In Proceedings of the 2009 ACM workshop on Cloud computing security, pp. 103-114. ACM, 2009.

[55] Institute of Medicine (US) Committee on Health Research and the Privacy of Health Information: The HIPAA Privacy Rule; Nass SJ, Levit LA, Gostin LO, editors. Beyond the HIPAA Privacy Rule: Enhancing Privacy, Improving Health Through Research. Washington (DC): National Academies Press (US); 2009. 2, The Value and Importance of Health Information Privacy. Available from: https://www.ncbi.nlm.nih.gov/books/NBK9579/

[56] Robert Pear. "Privacy Issue Complicates Push to Link Medical Data" http://www.nytimes.com/2009/01/18/us/politics/18health.html on 25 Oct. 2016

[57] Tom Leek. "Pattern to allow multiple persons to decrypt a document, without sharing the encryption key?" http://security.stackexchange.com/a/71915 on 10 Oct. 2016

[58] Food and Drug Administration. "General Wellness:Policy for Low Risk Devices" http://www.fda.gov/downloads/medicaldevices/deviceregulationandguidance/guidancedocuments/ucm429674.pdf on 26 Oct. 2016

[59] Joanna Stern. "Fitness Bands With Heart-Rate Tracking Are Missing a Beat" http://www.wsj.com/articles/fitness-bands-with-heart-rate-tracking-are-missing-a-beat-1418759929 on 26 Oct. 2016

[60] Stahl SE, An H-S, Dinkel DM, et al. How accurate are the wrist-based heart rate monitors during walking and running activities? Are they accurate enough?. BMJ Open Sport Exerc Med 2016;2:e000106. doi:10.1136/bmjsem-2015- 000106

[61] Rod Trent. “Scientifically Speaking, Which Fitness Device is Most Accurate?” http://winsupersite.com/microsoft-band/scientifically-speaking-which-fitness-device-most-accurate on 16 Sep. 2016

[62] Shane Richmond. “The real world wrist-based heart rate monitor test: Are they accurate enough?” http://www.wareable.com/fitness-trackers/heart-rate-monitor-accurate-comparison-wrist on 16 Sep. 2016

[63] Mike Prospero. “Who Has The Most Accurate Heart Rate Monitor?” http://www.tomsguide.com/us/heart-rate-monitor,review-2885.html on 16 Sep. 2016

[64] NeuroSky. “ECG vs PPG for Heart Rate Monitoring: Which is Best?” http://neurosky.com/2015/01/ecg-vs-ppg-for-heart-rate-monitoring-which-is-best/ on 16 Sep. 2016

[65] NeuroSky. “Electrical (ECG) vs. Optical-based (PPG) Biosensors in Wearable Devices” http://neurosky.com/wp-content/uploads/2016/06/TOF-side-by-side-competitor-comparison.pdf on 16 Sep. 2016

[66] Arnold JM, Fitchett DH, Howlett JG, Lonn EM, Tardif J-C. Resting heart rate: A modifiable prognostic indicator of cardiovascular risk and outcomes? The Canadian Journal of Cardiology. 2008;24(Suppl A):3A-8A.

[67] Yu-quian Li, Chang-qing Sun, Lin-lin Li, Ling Wang, Yi-rui Guo, Ai-gui You, Yuan-lin Xi, Chong-jian Wang. Resting heart rate as a marker for identifying the risk of undiagnosed type 2 diabetes mellitus: a cross-sectional survey. Bio Med Central Public Health. 2014; 14:1052 DOI: 10.1186/1471-2458-14-1052

[68] Margaret Rouse. “Business Logic”
http://whatis.techtarget.com/definition/business-logic

[69] Margaret Rouse. “Use Case”
http://searchsoftwarequality.techtarget.com/definition/use-case

[70] Eric G. “What are the differences between dictionary attack and brute force attack?” http://security.stackexchange.com/a/67768

[71] Evan Sharp. “AngularJS: Difference between angular-route and angular-ui-router” http://stackoverflow.com/questions/21023763/angularjs-difference-between-angular-route-and-angular-ui-router

[72] W3Schools.com. “AngularJS AJAX - $http”
http://www.w3schools.com/angular/angular_http.asp

Biographical Information

The author has been a student of the field of computer science for more than 6 years at the time of this writing. He received his Master of Science Degree in Computer Science and Engineering from The University of Texas at Arlington in December 2016. This was preceded by a Bachelor of Engineering Degree in Computer Science and Engineering from Rajiv Gandhi Technical University, Bhopal, India in June 2014.

Author has keen interest in emerging technologies, especially web, that help people and developers do better at what they want to do. His research interests include cloud computing, software engineering and software security. During his career as a graduate computer science student, author took part in various software projects such as “MavRide”, which is a ride dispatching and managing system made to ease and automate the process of security rides offered to university campus personnel. “MavQ”, that provides an alternative and a way to simplify, manage and automate the problem of long physical queues when seeking advice from a specific person such as an Academic Advisor.

The author has future plans of contributing to the society in a positive way by all means possible by leveraging the knowledge he has incurred as a student of computer science.