

Tracking The Influence of Wearables in Remote Patient Recovery

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in

INFORMATION TECHNOLOGY

By

VASAN S K (Reg No 38120089)

VINOTH V (Reg No 38120093)



**DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC I 12B Status by UGC I Approved by AICTE

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with "A" grade by NAAC | 12B Status by UGC | Approved by AICTE

Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai – 600 119

www.sathyabama.ac.in

SCHOOL OF COMPUTING

BONA FIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of **VASAN S K (Reg no:38120089)** and **VINOTH V (Reg no:38120093)** carried out the project entitled **“TRACKING THE INFLUENCE OF WEARABLES IN REMOTE PATIENT RECOVERY”** under our supervision from **OCTOBER 2021 to JANUARY 2022**

Internal Guide

Head of the Department

Dr. J. JABEZ M.E., Ph.D.,

Dr. R. SUBHASHINI M.E., Ph.D.,

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We, **VASAN S K and VINOTH V** hereby declare that the Project Report entitled done by me under the guidance of **Dr. J. JABEZ M.E., Ph.D.**, at Sathyabama Institute of Science and Technology (Deemed to be University), Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai - 600119 is submitted in partial fulfillment of the requirements for the award of Bachelor of Technology degree in Information Technology.

1.

2.

DATE:

PLACE: CHENNAI

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

We are pleased to acknowledge my sincere thanks to Board of management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey my thanks to **D. T. SASIKALA M.E., Ph.D., Dean, School of Computing** and **DR. R. SUBHASHINI M.E., Ph.D., Head of the Department, Department of Information Technology** for providing me the necessary support and details at the right time during the progressive reviews.

I would like to express our sincere and deep sense of gratitude to our Project Guide

Dr. J. JABEZ M.E., Ph.D., for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the Department of **INFORMATION TECHNOLOGY** who were helpful in many ways for the completion of the project.

ABSTRACT

Nowadays, consulting the doctor when the health of the patient goes abnormal, is something like the patient should go and explain what happened to the doctor and then doctor will do a diagnosis once again to confirm the abnormal state. On the other hand, using wearables will track the biological vitals of the person and will let that person know about his/her health in a impressive and simple UI. Considering these both scenarios, it is decided that to combine these two processes and to automate them in a software. I.e., We will read the vitals data from any wearable device's app (Mi Band App, Dozee App, etc.), And then we will try to access that data from our mobile app and will store it in database which will necessarily be noticed by the doctor. Basically, instead of tracking wearable data and showing only in person's mobile, it'll be stored in a database. So that, The related doctor of that particular person (i.e., patient). Wearable Health Devices (WDH), IOT and also software development are the better combinations of the modern world to detect health and fitness conditions and medical levels providing more data to healthcare workers such as doctors, front-desk peoples and surgeons. In olden days, people were measuring human vitals by holding wrist, etc for heart rate, pulse rate, etc. But now in modern days, due to technology improvement, we measured our vitals by inventions / modern equipment. For example, the heart beat is calculated by stethoscope, etc. These vitals are measured and manually entered into patient database in software product. And it's a time-consuming process to save the vitals details multiple times a day for each patient every day. Then, to overcome these issues, Wearable devices like smart watches are invented, which contain various sensors capable of measuring various human vitals. This project (module) will collect vitals from patient and records into Health Score (software product) patient database. So those data will be used by either patient or by doctor in case of any emergency or needed.

Keywords: front-desk, surgeons, heart rate, healthcare, software product

INDEX

CHAPTER NO	CHAPTER NAME	PAGE NO
	ABSTRACT	5
	LIST OF FIGURES	8
	LIST OF ABBREVIATION	9
1	INTRODUCTION	10
	1.1 GENERAL	10
	1.2 INTRODUCTION TO WEARABLES IN REMOTE PATIENT RECOVERY	11
2	LITERATURE SURVEY	12
	2.1 FOG COMPUTING-BASED SMART HEALTH MONITORING SYSTEM DEPLOYING LORA WIRELESS COMMUNICATION	12
	2.2 WEARABLE HEALTH DEVICES—VITAL SIGN MONITORING, SYSTEMS AND TECHNOLOGIES	12
	2.3 SMART WEARABLE SYSTEMS: CURRENT STATUS AND FUTURE CHALLENGES	13
	2.4 LEVERAGING KNOWLEDGE FROM PHYSIOLOGICAL DATA: ON-BODY HEAT STRESS RISK PREDICTION WITH SENSOR NETWORKS	14
	2.5 SMART WEARABLE DEVICES IN CARDIOVASCULAR CARE: WHERE WE ARE AND HOW TO MOVE FORWARD	14
	2.6 ACCURATE TEMPERATURE MEASUREMENTS FOR MEDICAL RESEARCH USING BODY SENSOR NETWORKS	15
	2.7 A SURVEY ON WEARABLE TECHNOLOGY: HISTORY, STATE-OF-THE-ART AND CURRENT CHALLENGES	15
	2.8 WIRELESS REMOTE PATIENT MONITORING ON GENERAL HOSPITAL WARDS	16
	2.9 A NEW TEXTILE-BASED WEARABLE DEVICE FOR BIOLOGICAL SIGNAL MONITORING. APPLICABILITY IN DAILY LIFE AND CLINICAL SETTING	16
	2.10 EXPLORING THE SMARTWATCH AS A TOOL FOR MEDICAL ADHERENCE	17

3	AIM AND SCOPE	18
	3.1 THE MAIN AIM OF THE PROJECT	18
	3.2 SCOPE OF TRACKING THE INFLUENCE OF WEARABLES IN REMOTE PATIENT RECOVERY	19
4	EXPREMENTAL OR MATERIAL METHODS ALGORITHM	21
	4.1 HOW DO MODULES WORK?	22
	4.2. LIST OF MODULES	22
	4.3 EXPLANATION AND WORKING	23
	4.4 WHATS USEFUL?	25
	4.5 METHODOLOGIES	25
5	RESULT AND DISCUSSION	29
	5.1 RESULT	29
	5.2 DISCUSSION	29
6	SUMMARY AND CONCLUSION	31
	REFERENCE	33
	APPENDIX	
	A. SOURCE CODE	35
	B. SCREEN SHOTS	53

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
	The overall process behind the project	20
3.1	Architecture Diagram	24
4.1	Authorize Screen	57
5.1	Choosing Account Screen	58
5.2	Vitals Screen	59
5.3	Heartrates Day vice screen	60
5.4	Heartrates Month vice Screen	61
5.5	Step Tracking week vise screen	62
5.6	Step Tracking year vise screen	63
5.7		

LIST OF ABBREVIATIONS

ACCRONYM

FDA
ECG
RPM
UHS
SWS
HM
WHD
IoT
UI
LoRa

ABREVIATION

Food and Drug Administration
Electrocardiogram
Remote Patient Monitoring
Uncompensable Heat Stress
Smart Wearable Systems
Health Monitoring
World Health Devices
Internet of Things
User Interface
Long Range

CHAPTER 1

INTRODUCTION

1.1 GENERAL

Wearable Health Devices (WHDs) are an emerging technology that enables continuous ambulatory monitoring of human vital signs during daily life (during work, at home, during sport activities, etc.) or in a clinical environment, with the advantage of minimizing discomfort and interference with normal human activities.

WHDs are part of personal health systems, a concept introduced in the late 1990s, with the purpose of placing the individual citizen in the center of the healthcare delivery process, managing its own health and interacting with care providers—a concept that is commonly referred to as “patient empowerment. The aim was to raise people interest about their health status, improving the quality of care and making use of the new technology capabilities. These devices create a synergy between multiple science domains such as biomedical technologies, micro and nanotechnologies, materials engineering, electronic engineering and information and communication technologies.

The use of WHDs allows the ambulatory acquisition of vital signs and health status monitoring over extended periods (days/weeks) and outside clinical environments. This feature allows acquiring vital data during different daily activities, ensuring a better support in medical diagnosis and/or helping in a better and faster recovering from a medical intervention or body injury. WHDs are also very useful in sport activities/fitness to monitor athlete’s performance or even in first responders or military personnel to evaluate and monitor their body response in different hazardous situations and to better manage their effort and occupational health. These devices can be for both medical and/or activities/fitness/wellness purposes, always targeting the human body monitoring. Taking this in account, the best terminology is “health”, leading to WHDs. WHDs denomination can be more specific referring to which areas they are applied to. Independently of WHDs purpose, there are four main requirements on their design: low power consumption, reliability and security, comfort and ergonomics.

1.2 INTRODUCTION TO WEARABLES IN REMOTE PATIENT RECOVERY

According to Statista, the wearable devices market is currently having a worldwide revenue of around \$26 billion, and is expected to reach almost \$34 billion in 2019. Regarding healthcare and medical environments, it is expected to grow almost to \$15 billion worldwide value in 2019. This review aims to gather recent information on WHDs and better evaluate the current situation of such devices, foreseeing their evolution in the coming years. The main focus will be in vital signs and in textile embedded WHDs.

The document is organized in seven sections. we discuss what really are the most needed measurements to be acquired concerning medical, personal healthcare, fitness/wellness and sport activities areas. The signs where a higher technological development effort is being made is also covered. It presents recent technologies used to acquire each identified vital sign. A generic system architecture is presented to better understand WHDs components, workflow and the differences between existing devices. It reviews some types of WHD with a main focus on heart activity monitoring devices. A focus on commercialized WHDs t-shirts is made to compare devices and discuss properties. Some prototypes are also briefly presented. To foresee possible future trends in this area, this review is complemented with a market analysis in Section 6. Finally, we conclude this review referring to some future challenges and perspectives in the WHDs area. For example, in recent publications there is a very good description of the technical aspects related with the physiologic signs description, acquisition methods, some devices and fabrication methods, but few information about the concerns and needs of those signs on the human health or about the WHDs system architecture. On the other hand, some other publications are strongly focused on the WHDs architecture and technical aspects of it, such as in There are still other publications that focus on other aspects, but here we present a concatenation of the main aspects related with WHDs. This review also addresses two topics that are not very much explored in WHDs technology publications which are the understanding of the most important vital signs, where some medical aspects are included, and also a market trend analysis to understand what can be the future of these type of devices. A unique characteristic of this review is the heart activity monitoring focus that is performed to understand the state-the-art of one of the largest areas of WHDs.

CHAPTER 2

LITERATURE SURVEY

2.1 FOG COMPUTING-BASED SMART HEALTH MONITORING SYSTEM DEPLOYING LORA WIRELESS COMMUNICATION

Health is a major aspect of life and determines the quality of life. In addition, healthy citizens are more productive and directly influence a country's economy. Thus, for citizens to not get deprived of primary care, it is important to have a proper health monitoring system. For creating such health monitoring system, making use of Internet of Things (IoT) can be the best solution. Fog computing services can be provided to IoT by creating local servers and making use of some computers. Various wireless connectivity solutions being developed have finally given IoT the ability to communicate with each other easily. One such connectivity solution is the Long Range (LoRa) radio which lets IoT communicate over long range with low energy consumption. In this paper, an architecture for smart health monitoring system is proposed and is implemented by creating a basic testbed. The system is an application of IoT deploying LoRa wireless communication under fog computing, benefitting the system with both the qualities of LoRa and fog computing. The test results show that the proposed system can be promising for changing the clinic-centric health system to smart patient-centric health system and for providing seamless health services to all.

2.2 WEARABLE HEALTH DEVICES—VITAL SIGN MONITORING, SYSTEMS AND TECHNOLOGIES

Wearable Health Devices (WHDs) are increasingly helping people to better monitor their health status both at an activity/fitness level for self-health tracking and at a medical level providing more data to clinicians with a potential for earlier diagnostic and guidance of treatment. The technology revolution in the miniaturization of electronic devices is enabling to design more reliable and adaptable wearables, contributing for a world-wide change in the health monitoring approach. In this paper we review important aspects in the WHDs area, listing the state-of-the-art of wearable vital signs sensing technologies plus their system architectures and specifications. A focus on vital signs acquired by WHDs is made: first a discussion about the most important vital signs for health assessment using WHDs is presented and then for each vital sign a description is

made concerning its origin and effect on health, monitoring needs, acquisition methods and WHDs and recent scientific developments on the area (electrocardiogram, heart rate, blood pressure, respiration rate, blood oxygen saturation, blood glucose, skin perspiration, capnography, body temperature, motion evaluation, cardiac implantable devices and ambient parameters). A general WHDs system architecture is presented based on the state-of-the-art. After a global review of WHDs, we zoom in into cardiovascular WHDs, analyzing commercial devices and their applicability versus quality, extending this subject to smart t-shirts for medical purposes. Furthermore, we present a resumed evolution of these devices based on the prototypes developed along the years. Finally, we discuss likely market trends and future challenges for the emerging WHDs area.

2.3 SMART WEARABLE SYSTEMS: CURRENT STATUS AND FUTURE CHALLENGES

Extensive efforts have been made in both academia and industry in the research and development of smart wearable systems (SWS) for health monitoring (HM). Primarily influenced by skyrocketing healthcare costs and supported by recent technological advances in micro- and nanotechnologies, miniaturization of sensors, and smart fabrics, the continuous advances in SWS will progressively change the landscape of healthcare by allowing individual management and continuous monitoring of a patient's health status. Consisting of various components and devices, ranging from sensors and actuators to multimedia devices, these systems support complex healthcare applications and enable low-cost wearable, non-invasive alternatives for continuous 24-h monitoring of health, activity, mobility, and mental status, both indoors and outdoors. Our objective has been to examine the current research in wearable to serve as references for researchers and provide perspectives for future research. Herein, we review the current research and development of and the challenges facing SWS for HM, focusing on multi-parameter physiological sensor systems and activity and mobility measurement system designs that reliably measure mobility or vital signs and integrate real-time decision support processing for disease prevention, symptom detection, and diagnosis. For this literature review, we have chosen specific selection criteria to include papers in which wearable systems or devices are covered. We describe the state of the art in SWS and provide a survey of recent implementations of wearable health-care systems. We describe current issues, challenges, and prospects of SWS.

2.4 LEVERAGING KNOWLEDGE FROM PHYSIOLOGICAL DATA: ON-BODY HEAT STRESS RISK PREDICTION WITH SENSOR NETWORKS

The paper demonstrates that wearable sensor systems, coupled with real-time on-body processing and actuation, can enhance safety for wearers of heavy protective equipment who are subjected to harsh thermal environments by reducing risk of Uncompensable Heat Stress (UHS). The work focuses on Explosive Ordnance Disposal operatives and shows that predictions of UHS risk can be performed in real-time with sufficient accuracy for real-world use. Furthermore, it is shown that the required sensory input for such algorithms can be obtained with wearable, non-intrusive sensors. Two algorithms, one based on Bayesian nets and another on decision trees, are presented for determining the heat stress risk, considering the mean skin temperature prediction as a proxy. The algorithms are trained on empirical data and have accuracies of $92.1 \pm 2.9\%$ and $94.4 \pm 2.1\%$, respectively when tested using leave-one-subject-out cross-validation. In applications such as Explosive Ordnance Disposal operative monitoring, such prediction algorithms can enable autonomous actuation of cooling systems and haptic alerts to minimize casualties.

2.5 SMART WEARABLE DEVICES IN CARDIOVASCULAR CARE: WHERE WE ARE AND HOW TO MOVE FORWARD

Technological innovations reach deeply into our daily lives and an emerging trend supports the use of commercial smart wearable devices to manage health. In the era of remote, decentralized and increasingly personalized patient care, catalysed by the COVID-19 pandemic, the cardiovascular community must familiarize itself with the wearable technologies on the market and their wide range of clinical applications. In this Review, we highlight the basic engineering principles of common wearable sensors and where they can be error-prone. We also examine the role of these devices in the remote screening and diagnosis of common cardiovascular diseases, such as arrhythmias, and in the management of patients with established cardiovascular conditions, for example, heart failure. To date, challenges such as device accuracy, clinical validity, a lack of standardized regulatory policies and concerns for patient privacy are still hindering the widespread adoption of smart wearable technologies in clinical practice. We present several recommendations to navigate these challenges and propose a simple and practical

‘ABCD’ guide for clinicians, personalized to their specific practice needs, to accelerate the integration of these devices into the clinical workflow for optimal patient care.

2.6 ACCURATE TEMPERATURE MEASUREMENTS FOR MEDICAL RESEARCH USING BODY SENSOR NETWORKS

Medical measurements and clinical trials are often carried out in controlled lab settings – severely limiting the realism and duration of such studies. Our goal is henceforth to design a body sensor network for unobtrusive and highly accurate profiling of body parameters over weeks in realistic environments. One example application is monitoring the impact of sleep deprivation on periodic processes in the human body known as circadian rhythms, which requires highly accurate profiling of skin temperature across the human body over weeks with real-time feedback to a remote medic. We analyze the requirements on a body sensor network for such applications and highlight the need for self-organizing behavior such as adaptive sampling to ensure energy efficiency and thus longevity, adaptive communication strategies, self-testing, automatic compensation for environmental conditions, or automatic recording of a diary of activities. As a first step towards this goal, we design and build a prototype of such a non-invasive wearable wireless monitoring system for accurate body temperature measurements and real time feedback to the medic. Through the design, parameterization, and calibration of an active measurement subsystem, we obtain an accuracy of 0.02°C Over the typical body temperature range of $16\text{-}42^{\circ}\text{C}$. We report results from two preliminary trials regarding the impact of circadian rhythms and mental activity on skin temperature, indicating that our tool could indeed become a valuable asset for medical research.

2.7 A SURVEY ON WEARABLE TECHNOLOGY: HISTORY, STATE-OF-THE-ART AND CURRENT CHALLENGES

Technology is continually undergoing a constituent development caused by the appearance of billions new interconnected “things” and their entrenchment in our daily lives. One of the underlying versatile technologies, namely wearables, is able to capture rich contextual information produced by such devices and use it to deliver a legitimately personalized experience. The main aim of this paper is to shed light on the history of wearable devices and provide a state-of-the-art review on the wearable market. Moreover, the paper provides an extensive and diverse classification of wearables, based

on various factors, a discussion on wireless communication technologies, architectures, data processing aspects, and market status, as well as a variety of other actual information on wearable technology. Finally, the survey highlights the critical challenges and existing/future solutions.

2.8 WIRELESS REMOTE PATIENT MONITORING ON GENERAL HOSPITAL WARDS

A novel approach which has potential to improve quality of patient care on general hospital wards is proposed. Patient care is a labour - intensive task that requires high input of human resources. A Remote Patient Monitoring (RPM) system is proposed which can go some way towards improving patient monitoring on general hospital wards. In this system vital signs are gathered from patients and sent to a control unit for centralized monitoring. The RPM system can complement the role of nurses in monitoring patients' vital signs. They will be able to focus on holistic needs of patients thereby providing better personal care. Wireless network technologies, ZigBee and Wi-Fi, are utilized for transmission of vital signs in the proposed RPM system. They provide flexibility and mobility to patients. A prototype system for RPM is designed and simulated. The results illustrated the capability, suitability and limitation of the chosen technology.

2.9 A NEW TEXTILE-BASED WEARABLE DEVICE FOR BIOLOGICAL SIGNAL MONITORING. APPLICABILITY IN DAILY LIFE AND CLINICAL SETTING

The paper presents a new textile-based wearable system for the unobtrusive recording of cardiorespiratory and motion signals during spontaneous behavior along with the first results concerning the application of this device in daily life and in a clinical environment. The system, called MagIC (Maglietta Interattiva Computerizzata), is composed of a vest, including textile sensors for detecting ECG and respiratory activity, and a portable electronic board for motion detection, signal preprocessing and wireless data transmission to a remote monitoring station. The MagIC system has been tested in freely moving subjects at work, at home, while driving and cycling and in microgravity condition during a parabolic flight. Applicability of the system in cardiac in-patients is now under evaluation. Preliminary data derived from recordings performed

on patients in bed and during physical exercise showed 1) good signal quality over most of the monitoring periods, 2) a correct identification of arrhythmic events, and 3) a correct estimation of the average beat-by-beat heart rate. These positive results support further developments of the MagIC system, aimed at tuning this approach for a routine use in clinical practice and in daily life

2.10 EXPLORING THE SMARTWATCH AS A TOOL FOR MEDICAL ADHERENCE

Adherence to medication is generally described as a huge problem in the healthcare system. The term adherence is generally preferred by many health care providers as the word 'compliance' describes a patient who is passively taking medication as advised/ordered by the doctor. This thesis goes in depth in identifying the problems faced to achieve maximum adherence to medication and the important factors contributing to it. The objective is to come up with an alternative approach to help improve medical adherence using a smart watch-based application that reminds patients to consume their medicines in a timely fashion. It addresses precisely which medication to take and in what quantity. This form of reporting and alerting is believed to achieve higher levels of adherence based on grounded theory. Shedding light on the methodologies used while clearly identifies the assumptions and limitations such a system.

CHAPTER 3

AIM AND SCOPE

Remote patient monitoring is about creating a better quality of life for patients, whether they suffer from chronic illnesses or are recovering from one-time injuries. While initially limited in its applications, RPM use cases are evolving rapidly, thanks partially to advances in wearable technology, which are making monitoring and tracking of various health-related statistics easier and more consistent.

Biosensors have been integrated into many form factors, including watches and other wristwear, patches, shoes, belts, clothing, and of course smartphones. These devices come with several benefits to both patients and physicians – the two key constituencies to any monitoring and treatment program – and are being used to treat a growing number of conditions.

3.1 THE MAIN AIM OF THE PROJECT

- The main aim of the project is, particularly in remote patient management and delivery of care across the healthcare ecosystem, wearable technology, and product maturation have enabled providers with real-time data collection, increasing accuracy and informing decision making. Wearables have the potential to improve patient outcomes by monitoring physical health, equipping providers with data they otherwise may not have access to. Miniaturizing ECG, Blood Pressure, Heart Rate, etc. monitoring - these devices continue to disrupt traditional tools of patient data capture.

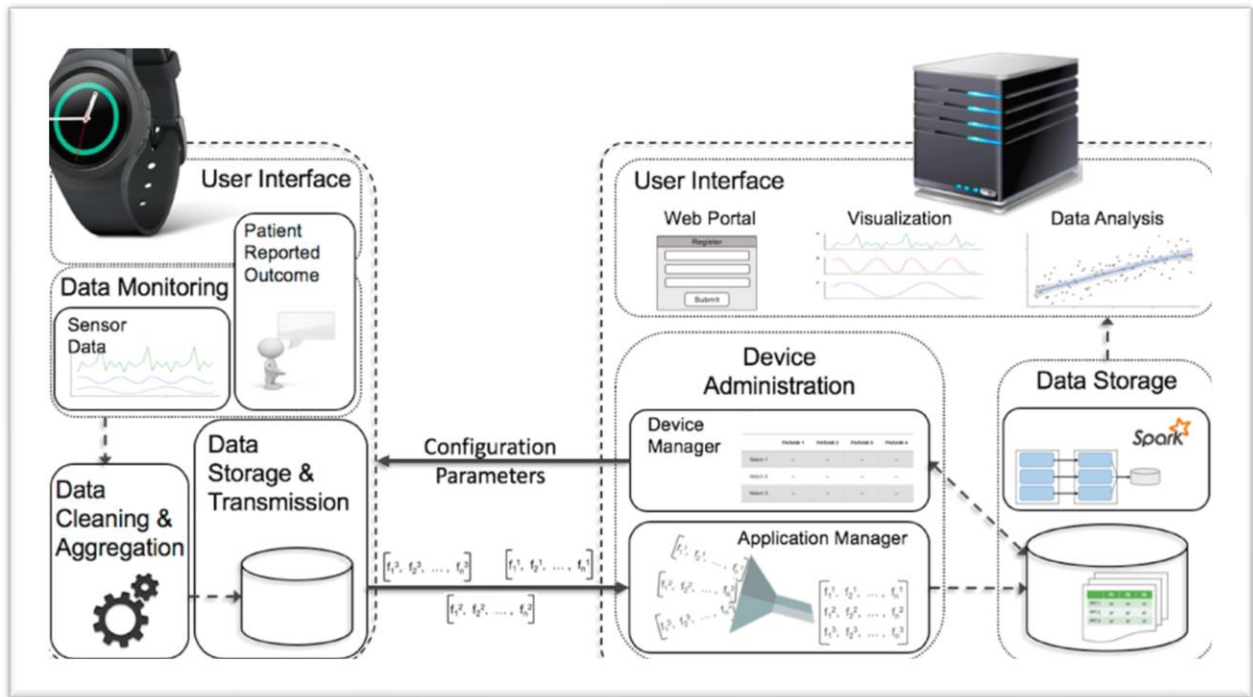


Figure 3.1- The overall process behind the project

3.2 SCOPE OF TRACKING THE INFLUENCE OF WEARABLES IN REMOTE PATIENT RECOVERY

3.2.1. Managing Chronic Conditions

Managing many conditions can feel overwhelming to patients and physicians, both of whom have been involved in manually tracking, recording, and analyzing statistics like heart rate, blood glucose levels, body temperature and more. Wearable devices make it possible to gather the same data automatically, send it to telehealth systems and physicians for analysis and automatic appended to patients' EMRs, keeping patient data accurate and up-to-date without any manual data entry. Systems can also be set up to send alerts for treatment updates if certain thresholds are exceeded. The end result is the data collection simply becomes a passive task for everyone involved, allowing them to focus on other activities.

For instance, diabetics are able to use continuous glucose monitoring transmitters that connect to smartphone apps to provide alerts if blood sugar levels are too low or high. Ongoing data collection can also help physicians and patients understand how different foods or activities impact their blood sugar levels and create appropriate action plans.

3.2.2. Wellness Programs

Exercise is often prescribed for many chronic and temporary conditions, including weight control, anxiety or depression, and injury rehabilitation. Fitness wearables are ideally suited to track key statistics that help track effectiveness of a prescribed regimen, while additional devices, like connected scales, make it possible to also measure related statistics, like weight. In addition to measuring heart rate and other vitals, these wearables act as reminders for patients who may otherwise forget to follow their programs, or provide motivation for patients to achieve certain fitness-related goals. Again, the data can be shared directly with physicians and automatically inserted into patient records. Wearable devices with accelerometers can be used in rehabilitation after surgery or injury and data can be shared in real time with trainers who can provide immediate feedback.

3.2.3. Mental Health and Cognitive Challenges

Patients with Alzheimer's disease, mental health issues, or other cognitive challenges can use wearable devices that act as trackers to allow caregivers to monitor patients' location and, depending on the specific device, even check in with them. Some devices can even use geofencing technology to trigger call, texts or vibration alerts if patients venture too far from their homes. Devices can also deliver reminders for medication or appointments. The end results is a more effective way of monitoring patient location and providing reminders to follow prescribed protocols, making treatment more effective.

3.2.4. Emerging Innovation and the Future of Healthcare Wearables

While there are already many common conditions that can leverage non-intrusive wearables to improve or simplify treatment, new, innovative use cases are constantly being developed that can treat, monitor, and even prevent certain conditions. For instance, an FDA-cleared wrist device is able to prevent nausea from motion sickness and even morning sickness during pregnancy through neuromodulation. Another company is working on embedded breast patches in bras that detect changes in breast tissue and act as screening devices for breast cancer.

Preventative monitoring may be a significant growth opportunity, as applications for monitoring at-risk or even low-risk patients has the potential to transform healthcare from a management to a preventative wellness protocol using RPM and wearables.

The true success of wearables as part of remote patient monitoring solutions will be determined on two levels: benefits to patients and physicians. But, as long as the medical outcomes are at least consistent with office visits, the time- and cost- benefits for both will drive adoption. For patients, the convenience of not having to travel to offices and reduced healthcare costs, along with the potential for expedited.

CHAPTER 4

EXPREMENTAL OR METHODS

Smartphone and smartwatch technology is changing the transmission and monitoring landscape for patients and research participants to communicate their healthcare information in real time. Flexible, bidirectional and real-time control of communication allows development of a rich set of healthcare applications that can provide interactivity with the participant and adapt dynamically to their changing environment. Additionally, smartwatches have a variety of sensors suitable for collecting physical activity and location data.

The combination of all these features makes it possible to transmit the collected data to a remote server, and thus, to monitor physical activity and potentially social activity in real time. As smartwatches exhibit high user acceptability and increasing popularity, they are ideal devices for monitoring activities for extended periods of time to investigate the physical activity patterns in free-living condition and their relationship with the seemingly random occurring illnesses, which have remained a challenge in the current literature.

Therefore, the purpose of this study was to develop a smartwatch-based framework. Wearable sensors have been widely used in medical sciences, sports and security. Wearable sensors can detect abnormal and unforeseen situations, and monitor physiological parameters and symptoms through these trackers. This technology has transformed healthcare by allowing continuous monitoring of patients without hospitalization. Medical monitoring of patients' body temperature, heart rate, heart rate variability, brain activity, muscle motion and other critical data can be delivered through these trackers. Moreover, in sports training there is an increasing demand for wearable sensors.

4.1. HOW DO MODULES WORK?

Modules in this project were planned in such a way that, every step involved in this entire project is handled by a separate module. For example, implementing connection between client and server apps, retrieval of data from the server into the client and presenting the user interface to the user in such a way that it's easier to understand and useful.

4.2. LIST OF MODULES

First of all, this project itself is a module. In this, we have several sub-modules. The following are the list of modules.

1. Implementing connection b/w Google Fit and React Native.
2. Retrieve vitals data form Google Fit.
3. Present UI to the user.

4.2.1. Implementing connection b/w Google Fit and React Native.

This sub module takes care of connecting client mobile app (React Native) app with Google Fit API (Application Programmable Interface). Besides, this includes authenticating user against Google Fit, implementing a secured connection, etc. We will discuss the technical information of this module in the later topics.

4.2.2. Retrieve vitals data form Google Fit.

This module takes care of retrieving vitals data from Google Fit APIs. For example, if one user after authentication, requests for his/her heart beat data, this module will go and fetch those data from Google Fit APIs.

4.2.3. Present UI to the user.

Finally, this module takes care of showing the collected data in a user Interface. i.e., whenever, user requests, heart rate data, above mentioned module will fetch it and this module will show those data as graph in UI.

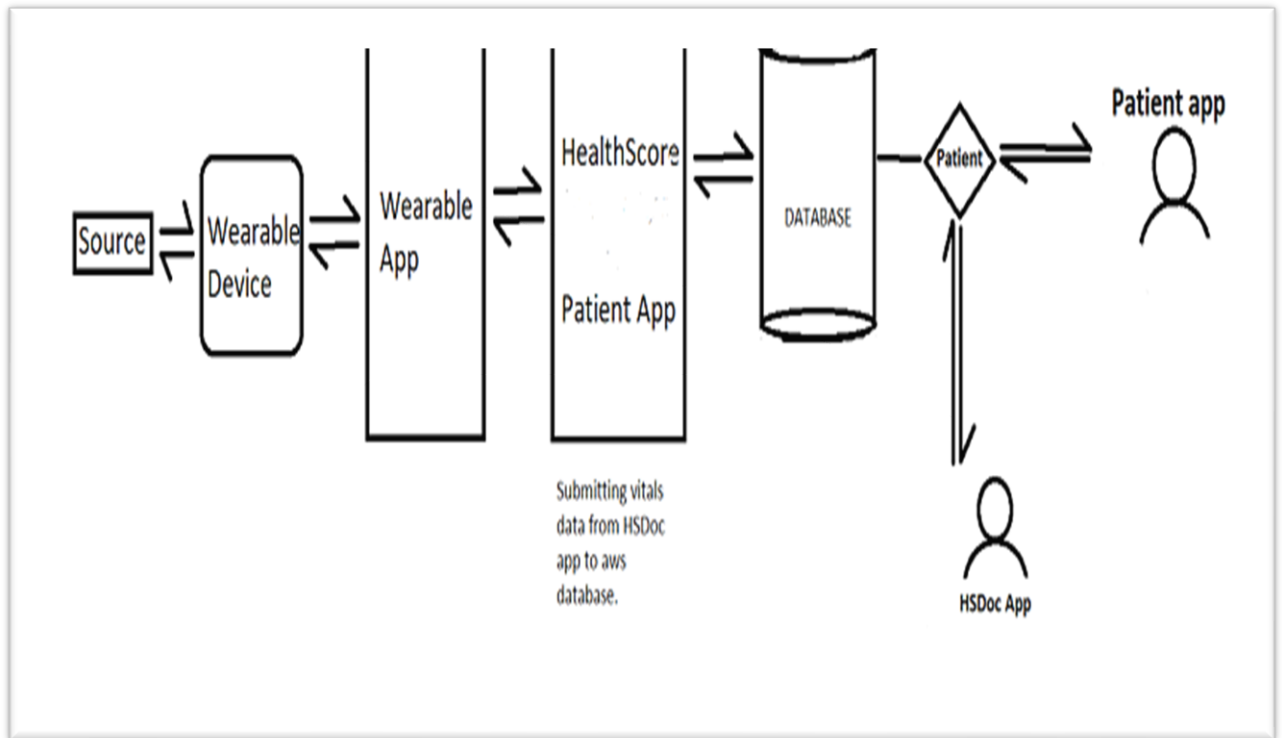


Fig 4.1- Architecture Daigram

4.3 EXPLANATION AND WORKING

Here in this section, We will see all the modules mentioned above in details in both technical and theoritical manner.

4.3.1. Implementing connection b/w Google Fit and React Native.

First of all, Let's talk a little bit about React Native and Google Fit. So that It's easy to understand the functionality of this module. And also, this section includes information like setting up the code level modules and using them.

Basically, we will have a React Native project which is a mobile app that is capable of running in both android and iOS. And also, Google Fit is an API or we can say it as server which is capable of accepting request from the client and give a response back to the client. Google Fit server is a service provided by Google. So, it has several inbuilt functionalities. Authentication is a good example for that. From this, we are able to understand that, authentication is the first step. Of course, without being authenticated, no data can be retrieved from the Google Fit. In order to implement authentication, following are the steps.

1. Create a google cloud project.

- a. Go to google clout console. And then create a new project.
2. Enable fitness API
 - a. Google provides many APIs to developers to make use of. Among them, fitness API is the one that we'll be making use of.
 - b. Go ahead and enable that API.
 - c. Then, go to the OAuth consent screen, and add several email addresses.
 - d. These email address will only be able to test this module / access Google Fit. This step is only for development stage. And not for production stage.
3. Install React Native Google Fit dependency
 - a. Google Fit dependency in the React Native project must be installed.
 - b. After installing that, Authentication callback must be invoked, which will authenticate user against Google Fit with user's Gmail account.

4.3.2. Retrieve vitals from Google Fit

1. After successful authentication, Google Fit internally (by Itself), keeps track of users who are authenticated and their session data.
2. Thus, any authenticated user can now access their own vitals data from the Google Fit API. As we know already, This is a Native mobile app. So, API calls will be done through pre-defined callbacks. Not using URLs.
3. Here, As we discussed previously itself, the package we are suing to connect React Native and Google Fit is called as react-native-google-fit.
4. This package supports many callbacks those which are handling API calls through URLs internally. Developer should invoke the callbacks with right payloads.
5. Following are several example callbacks.

5.1. await Google Fit.getHeartRateSamples({})

4.4 WHATS USEFUL?

- A. Reduced cost and time
- B. Secured and Risk-free conditions
- C. Efficient diagnosis based on captured vitals
- D. Insight into remote recovery with wearables

4.4.1. REDUCED COST AND TIME

Handling the same scenario is very and both time-consuming and cost spending works. As we are working in an environment that is most likely virtual environment, It leads us for reduced cost and time consumption.

4.4.2. SECURED AND RISK-FREE CONDITIONS

In case of creation of any technology or modules, That must have crossed the development stage which almost is as a virtual environment. It's always known that the development is a great environment / condition in which many real time scenarios can be tested. Thus, the development stage gives developers a great way to test all scenarios even before making actual changes.

4.4.3. EFFICIENT DIAGNOSIS BASED ON CAPTURED VITAL

In some scenarios like in an emergency / critical states of the patient, the diagnosis becomes easier and more efficient than the traditional way.

4.4.4. INSIGHT INTO REMOTE RECOVERY WITH WEARABLES

Unlike other existing methods, this vital capturing method allows the users (patients or doctors) to track the health of patient in time based manner from low level to well detailed level.

4.5 METHODOLOGIES

Methods proposed for this modules are,

1. Vitals capturing
2. Transfer to app, server and to database
3. Data analyzing

4. Data monitoring
5. Visual representation
6. Real-time example

For this module, we are going to use several technologies from software development as follows,

1. Wearable device (A smart watch to capture human vitals)
2. React native (A cross platform mobile app development framework)
3. AWS (Amazon Web Services to store the vitals data in database)

4.5.1. VITALS CAPTURING

Vitals are captured from various sensors present in wearable devices like smart watch. There are several vital we're going to need for this module as follows,

- a) Body Temperature – Temperature sensor
- b) Blood Pressure – Mercury Sphygmomanometer
- c) Blood Glucose State - Glucometer
- d) Resting Heart Rate – Heart Beat Sensor
- e) SPO2 Oximeter
- f) Respiratory Rate - Flowmeter
- g) Body Weight
- h) Body Height

These are the several human vitals those are going to be measured.

4.5.2. TRANSFER DATA TO THE APP, SERVER AND TO DATABASE

Those measured vitals are passed to and received by React Native app. After receiving the data, We will be transferring those data to the patient app server of the Health Score software product. And then the server will be saving those data in to AWS database bucket. At this point, we'll be ready to serve those data to the client-side. Thus, It'll be analyzed by the doctor or by the patient when needed.

4.5.3. DATA ANALYZING

Before saving those data, those must be analyzed at the time receiving them in the server. There several validations are to be done on those incoming vitals data, such as heart rate must be an integer value and with the minimum and maximum of particular.

4.5.4. DATA MONITORING

The ROAMM application collects patient-reported and sensor monitor data. As explained previously, patient-reported data are obtained by prompting the user questions about pre-defined outcomes, such as pain and fatigue. The Data Monitoring module interfaces with the Human Activity Monitor libraries that Tizen provides to collect sensor monitor data. Since accelerometers are widely used for activity monitoring and assessment, a validation of smartwatch's accelerometer data is warranted and has been the subject of recent works . Similarly, we compared the collected accelerometer data from the Samsung smartwatch with ActiGraph GT9X link to validate its measurements. Due to time drifts and processing shortcuts in smartwatches' accelerometer data and the fact that time points at which data are sampled are varying , we use normalized root mean squared values for each second of data from each axis. It shows that the smartwatch's accelerometers provide similar information as ActiGraph GT9X (correlation 0.97) for frequencies linked to human movement, i.e., 0.5–2.5 Hz. Thus, we consider the smartwatch a valid and accurate replacement of specialized devices for monitoring physical activities.

4.5.5. VISUAL REPRESENTATION

Thus, After doing validation and saving those data successfully into the AWS bucket. Then if one patient comes at a critical stage to the hospital without any past Report Summery, Then, at that scenario, the data from AWS buckets, which are stored previously by the patient's wearable device, are retrieved and will be shown in the mobile app user interface visually.

4.5.6. REAL-TIME EXAMPLE

Assume that a patient goes to an hospital in a critical stage by heart problem. After reaching the hospital, the patient will be diagnosed and treated by the doctors and then recovered. After the recovery of the patient, after few days of the bed rest, The

patient will be discharged. While being discharged patient will get the report summery with all the details such as,

- a) Health condition of that patient before coming to hospital.
- b) Health condition of that patient that caused that patient to come to that hospital.
- c) The treatment details those are provided by the hospital to the patient in order to recover the patient.
- d) Then, the Health condition of that patient that is expected to be, after being discharged.

Now, after discharge, assume that after few months, the patient's health is critical and is supposed to go to the same hospital as soon as possible. And also assume that the doctor who diagnosed the patient in his last visit and operated is not available. At this scenario, of course that patient will have to be treated by another doctor who doesn't even know about the patient's past health conditions. So, It'll take some time to diagnose the issue with the help of the summery report that was provided by that hospital to the patient last time. But by using this module, as those vitals of that particular patient are stored in database already, It'll just fetch those things from AWS database, analyze and will show it to the doctor in a way which is understandable easily by that doctor using React Native. Note that, those vitals data from AWS will only be shown in the mobile app user interface after validation.

CHAPTER 5

RESULT AND DISCUSSION

5.1 RESULT

Smart watches exhibit the advantages of small form factor and can be wrapped on the wrist for daily wear. Although the reported use of smart watch applications for patients with chronic diseases appear promising, we found only one study focused on managing patients in critical or emergency conditions. In order for these devices to gain wide acceptance by health professionals, rigorous research on their accuracy, completeness and effect on workflow should be conducted before smart watch applications are integrated into clinical practice. User studies to investigate ideal functionality, user interface design, and usability for a variety of clinical and patient settings are needed. Further research is required to understand the impact of smart watch applications on clinical practice. Health monitoring with wearable devices is the recent reality in Virtual hospitality and healthcare. The main concept behind WHD's (Wearable Health Devices) are based on the integration with available biosensors which are helpful in human vitals measuring, intelligent data processing. In our module also, we'll be gathering the data (measured vitals data) from wearable device into our android native app. And in case of any emergency conditions of the patient (assuming all patient vitals details and patient's health history are recorded in a database Diagnose the patient sooner. Thus, the proposed module will be help full in real-time scenarios. And in future also, we'll be improving several features like showing the data in a more efficient and understandable way such as using graphs, histogram, etc in future

5.2 DISCUSSION

Wearable health devices are a recent reality in healthcare and still under development with the aim to be integrate in medical health systems. There are some personal monitoring devices on the market capable to provide instantaneous single-parameter assessment and transmission. But the main value potential of WHDs is to integrate several biosensors, intelligent processing and alerts, to support medical applications, while interacting with health providers, using some of the technology that is not yet available on the market, only in research. The existing devices are highly expensive and work wireless based, technology that is not available everywhere in many countries, such as France where many consumers do not have access to broadband internet as reported in the literature in 2012 [15]. This type of data communication leads to another problem: privacy—a problem that concern healthcare, in the

prevention of information, leading people to have lower confidence in these devices, resulting in ethical problems. This is one of the main barriers and the suggested solution is to create clear guidelines to providing privacy, confidentiality and proper use of electronic medical information. A way to increase security of these data can pass by the use of personal non-clonable ID biometric technology to secure data, such as fingerprint, iris and ECG waveform. All this effort will allow sharing secure data with a direct impact on societies, such as in China where WHDs may contribute to lower healthcare cost by introducing preventive healthcare strategies, cutting equipment and labour costs, eliminating unnecessary health services.

The next few years present a set of new interesting challenges in WHDs and to overtake them an adaption of these technologies should be made. It is important to study user requirements to develop an integrated approach in health and wellness services instead of devices and applications for single diseases. The development of such devices and its integration in an architecture of intelligent home services may have some main issues that require some attention: device efficiency, reliability and unobtrusiveness; privacy and ethical issues as already covered; legislation within and between states or countries; interoperability; end-user training to use WHDs; and social inclusion preventing users isolation, which can lead to a decrease of interaction with caregivers and physicians.

The good news is that the market seems to become more aware of WHDs possibilities and this will lead to more investment in R&D and better products to reach the market, hopefully enhancing the market growth in the coming years.

In the sensor technology area, there are three main aspects that will have particular attention: long-term stability; resiliency; and biocompatibility. In textile-based sensors, washing cycles can cause mechanical, chemical and heat degradation, so an effort must be made to develop more resistant sensors to overcome stress caused by extended periods of use. Smaller sensors are also another important requirement, being associated to the electronics development, to increase functionality and portability of WHDs. The sensitivity of wearable sensors is an important characteristic that should be carefully analysed. Biocompatibility must be highly considerate, covering sensors with antimicrobial or protective coating, preventing any potential toxicity of nanomaterials.

Wearable health devices are able to monitor a large number of vital signs of human body, from advance sensor supervision in the case of infant respiratory to fitness applications

or even soldiers on the battlefield. This particularity of WHDs creates a big excitement around this technology and many opportunities to continue its development. With new advances in new materials, electronics and telecommunication information technology, together with the entry of big multinational companies, such as Google, but also of small start-ups, WHDs are expected to overcome their challenges and enter in the consumer market with a higher impact in the upcoming years.

CHAPTER 6

SUMMARY AND CONCLUSION

Wearable devices will be used to track health and fitness conditions and medical levels for doctors, front desk peoples and surgeons. These data will be used by either patient or doctor in case of any emergency or needed. Using body sensors for on-body heat stress risk prediction, smart wearable devices can be used in cardiac care. Wearable health devices allow continuous ambulatory monitoring of human vital signs during daily life or in a clinical environment. They are part of a concept known as "patient empowerment" that aims to improve the quality of care. Wearable Health Devices allow ambulatory acquisition of vital signs and health status monitoring over extended periods and outside clinical environments. They can be for both medical and/or activities/fitness/wellness purposes and their main requirements are low power consumption, reliability and security, comfort and ergonomics.

This document presents a review of WHDs that focuses on the most important measurements to be acquired concerning medical, personal healthcare, fitness/wellness and sport activities areas, the signs where a higher technological development effort is being made, and some types of WHDs with a main focus on heart activity monitoring devices. RPM is about creating a better quality of life for patients, whether they suffer from chronic illnesses or recover from one-time injuries. Wearable technology is improving remote patient monitoring.

The main aim of the project is to monitor patient's health remotely, improve patient outcomes, and inform providers with real-time data collection. Wearable devices can collect and send data to telehealth systems that analyse it and send alerts to physicians and patients, keeping the data accurate and up-to-date without any manual data entry. Exercise is often prescribed for weight control, anxiety or depression, and injury rehabilitation. Fitness wearables can be used to measure key statistics, like heart rate and weight, and are

ideally suited to track effectiveness of a prescribed regimen. Patients with Alzheimer's disease, mental health issues, or other cognitive challenges can use wearable devices as trackers to allow caregivers to monitor their location and send reminders for medication or appointments. There are several wearable devices that improve or simplify treatment of common conditions. These include a wrist device that prevents nausea from motion sickness and morning sickness during pregnancy.

Benefits for both patients and physicians will be determined on two levels, including time- and cost-benefits. Smartphone and smartwatch technology is changing the transmission and monitoring landscape for patients and research participants to communicate their healthcare information in real time. Smartwatches have a variety of sensors suitable for collecting physical activity and location data, and thus, are ideal devices for monitoring physical activity and potentially social activity. The modules in the entire project were planned in such a way that, every step involved was handled by a separate module. This project is a module that includes several sub-modules, such as implementing connection between Google Fit and React Native, retrieving vitals data from Google Fit, and presenting UI to the user. Basically, we will have a mobile app that is capable of running in both android and iOS. It will have a Google Fit Server, and it will help us implement authentication. After creating a Google Cloud project, enable the fitness API, add email addresses to the developer's account, install the React Native Google Fit dependency, authenticate the user to the Google Fit API, and retrieve the users' vitals data. The development stage is a great environment that allows testing many scenarios even before making actual changes. We'll be transferring the data from React Native app to the software product's patient app server, and then saving those data in to an AWS database bucket. The application collects patient-reported and sensor monitor data from a smartwatch. The data is validated with the ACTi Graph GT9X link and shows that the smartwatch is a valid and accurate replacement of specialized devices for monitoring physical activities. After doing validation and saving the data successfully to AWS buckets, the data is retrieved to a mobile app. After being discharged, a patient will get the report with the health condition of the patient before coming to the hospital, and the health condition of the patient after being discharged. After discharge, if the patient's health is critical, then it will take some time for the doctor to diagnose the patient with the help of the summery report. By using this module, all vitals' data from the AWS database will be analyzed and shown to the doctor.

Smart watches exhibit the advantages of small form factor and are easily wrapped on the wrist. Research is required to understand the impact of smart watch applications on clinical practice. Health Monitoring with Wearable Devices are the recent reality in virtual hospitality and healthcare. Our proposed app gathers vitals data from wearable devices, and in case of any emergency conditions, diagnoses the patient sooner.

REFERENCE

1. Lymberis A.G.L. Wearable health systems: From smart technologies to real applications; Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society; New York, NY, USA. 30 August–3 September 2006; pp.
2. Yussuff V., Sanderson R. The World Market for Wireless Charging in Wearable Technology. IHS; Englewood, CO, USA: 2014.
3. Chan M., Esteve D., Fourniols J.Y., Escriba C., Campo E. Smart wearable systems: Current status and future challenges. *Artif. Intell. Med.* 2012
4. Gaura E., Kemp J., Brusey J. Leveraging knowledge from physiological data: On-body heat stress risk prediction with sensor networks. *IEEE Trans. Biomed. Circuits Syst.* 2013
5. Cunha J.P.S., Cunha B., Pereira A.S., Xavier W., Ferreira N., Meireles L. Vital-Jacket: A wearable wireless vital signs monitor for patients' mobility in cardiology and sports; Proceedings of the 2010 4th International Conference on Pervasive Computing Technologies for Healthcare; Munich, Germany. 22–25 March 201
6. Rifat Shahriyar M.F.B., Kundu G., Ahamed S.I., Akbar M. Intelligent Mobile Health Monitoring System (IMHMS) *Int. J. Control Autom.* 2009
7. Lukowicz P., Anliker U., Ward J., Troster G., Hirt E., Neufelt C. AMON: A wearable medical computer for high-risk patients; Proceedings of the Sixth International Symposium on Wearable Computers; Seattle, WA, USA. 10 October 2002
8. Farjadian A.B., Sivak M.L., Mavroidis C. SQUID: Sensorized shirt with smartphone interface for exercise monitoring and home rehabilitation; Proceedings of the

2013 IEEE International Conference on Rehabilitation Robotics (ICORR); Seattle, WA, USA. 24–26 June 2013

9. Custodio V., Herrera F.J., Lopez G., Moreno J.I. A review on architectures and communications technologies for wearable health-monitoring systems. *Sensors*. 2012
10. Nuubo—Wearable Medical Technologies. [(accessed on 21 April 2017)]; Available online: <http://www.nuubo.com/>
11. IHS Technology. Top Healthcare Technology Trend Predictions —IHS Medical Devices & Healthcare IT. IHS Markit; London, UK: 2014
12. Statista B.I. Wearable Device Market Value from 2010 to 2018 (in Million U.S. Dollars) Statista Inc.; New York, NY, USA: 2015.
13. Wearable technologies and its future applications https://www.researchgate.net/publication/275580004_WEARABLE_TECHNOLOGIES_AND_ITS_FUTURE_APPLICATIONS.
14. A Survey on Wearable Technology: History, State-of-the-Art and Current Challenges (2021) <https://doi.org/10.1016/j.comnet.2021.108074>

APPENDIX

A) SOURCE CODE

AuthroizeScreen.js

```
import React, { useEffect, useState } from 'react';
import { View, Text, Button } from 'react-native';
import googleFit from 'react-native-google-fit';
import GoogleFit, { Scopes } from 'react-native-google-fit';

export const options = {
  scopes: [
    Scopes.FITNESS_ACTIVITY_READ,
    Scopes.FITNESS_ACTIVITY_WRITE,
    Scopes.FITNESS_BODY_READ,
    Scopes.FITNESS_BODY_WRITE,
    Scopes.FITNESS_BLOOD_PRESSURE_READ,
    Scopes.FITNESS_BLOOD_PRESSURE_WRITE,
    Scopes.FITNESS_BLOOD_GLUCOSE_READ,
    Scopes.FITNESS_BLOOD_GLUCOSE_WRITE,
    Scopes.FITNESS_NUTRITION_WRITE,
    Scopes.FITNESS_SLEEP_READ,
    Scopes.FITNESS_HEART_RATE_READ,
    Scopes.FITNESS_HEART_RATE_WRITE,
  ],
};

const AuthroizeScreen = ({
  navigation
}) => {
  const [isAuthorized, setIsAuthorized] = useState(false);

  useEffect(() => {
    if(isAuthorized === true) {
      navigation.replace("VITALSELECTION")
    } else {
      authorize();
    }
  }, [isAuthorized])

  const authorize = async () => {
    const res = await googleFit.authorize(options)
    setIsAuthorized(old => res.success)
  }

  const onDisconenct = () => {
```

```

        console.log('onDisconnected');
        GoogleFit.disconnect()
    }

    return (
        <View style={{
            backgroundColor: '#efefef',
            justifyContent: 'center',
            alignItems: 'center',
            flex: 1
        }}>
            <Button
                title={"Authorize Google Fit"}
                onPress={authorize}
                color={"#000"}
            />

            { /* <Text>{'\n'}</Text>

            <Button
                title={"Disconnect Google Fit"}
                onPress={onDisconenct}
                color={"#000"}
            /> */ }
        </View>
    );
}

export default AuthroizeScreen;

```

HeartRateFilterScreen.js

```

import React, { useEffect, useLayoutEffect, useState } from 'react';
import { View, Text, StyleSheet, TouchableOpacity, Dimensions } from 'react-native';
import GoogleFit, { Scopes, BucketUnit } from 'react-native-google-fit'
import { Grid } from 'react-native-svg-charts';
import { LineChart, AreaChart } from 'react-native-svg-charts'
import * as shape from 'd3-shape';
import { Line, Svg } from 'react-native-svg';
import { YAxis } from 'react-native-svg-charts';
import { XAxis } from 'react-native-svg-charts';
import googleFit from 'react-native-google-fit';

```

```

import { BarChart } from 'react-native-chart-kit';

const MINIMUM_THRESHOLD = 60;
const MAXIMUM_THRESHOLD = 112;
const HeartRateFilterScreen = ({
  navigation
}) => {
  const [heartRateSamples, setHeartRateSamples] = useState([]);
  const [BPMValues, setBPMValues] = useState([]);
  const [datesValues, setDatesValues] = useState([]);
  const [currentInterval, setCurrentInterval] = useState();

  useEffect(() => {
    navigation.setOptions({
      headerRight: () => (
        <View style={{ flexDirection: 'row' }}>
          <TouchableOpacity
            style={{ height: '100%', justifyContent: 'center', paddingHorizontal: 10,
            }}
            onPress={() => {
              navigation.goBack();
            }}
          >
            <Text style={{ color: '#fff' }}>Back</Text>
          </TouchableOpacity>

          <TouchableOpacity
            style={{ height: '100%', justifyContent: 'center', paddingHorizontal: 10,
            }}
            onPress={() => {
              googleFit.disconnect()
              navigation.replace("AUTHORIZESCREEN")
            }}
          >

```

```

        >
        <Text style={{ color: '#fff' }}>Logout</Text>
      </TouchableOpacity>
    </View>
  )
})
}, [navigation])
const onBucketUnitPress = async (interval) => {
  setCurrentInterval(old => interval);
  setBPMValues(() => []);
  setHeartRateSamples(old => []);
  const today = new Date();
  const timestamp = new Date().getTime();
  const endDateTime = new Date(today.getTime());
  const startDateTime = new Date(timestamp - (interval * (24 * 60 * 60 * 1000)));
  console.log(startDateTime, '\n\t', endDateTime);
  const samples = await GoogleFit.getHeartRateSamples({
    startDate: startDateTime,
    endDate: endDateTime,
    bucketInterval: 1,
    bucketUnit: "DAY"
  })
  setHeartRateSamples(old => samples)
}
useEffect(() => {
  onBucketUnitPress(1);
}, [])
useEffect(() => {
  if(heartRateSamples.length > 0) {
    let BPMArr = [];
    let DateArr = [];
    heartRateSamples.map((item, index) => {

```

```

        BPMArr.push(item.value);
        DateArr.push(item.startDate)
    })
    setBPMValues(old => BPMArr);
    setDatesValues(old => DateArr);
}
}, [heartRateSamples])

return (
    <View style={ss.screen}>
        <View style={ss.pickerContainer}>
            <TouchableOpacity style={{ss.pickerItemContainer, { backgroundColor:
currentInterval === 1 ? "gold" : '#000' }}} onPress={() => onBucketUnitPress(1)}>
                <Text style={{ss.pickerTitle, { color: currentInterval === 1 ? "#000" : "#fff"
}}} >Day</Text>
            </TouchableOpacity>
            <TouchableOpacity style={{ss.pickerItemContainer, { backgroundColor:
currentInterval === 7 ? "gold" : '#000' }}} onPress={() => onBucketUnitPress(7)}>
                <Text style={{ss.pickerTitle, { color: currentInterval === 7 ? "#000" : "#fff"
}}} >Week</Text>
            </TouchableOpacity>
            <TouchableOpacity style={{ss.pickerItemContainer, { backgroundColor:
currentInterval === 28 ? "gold" : '#000' }}} onPress={() => onBucketUnitPress(28)}>
                <Text style={{ss.pickerTitle, { color: currentInterval === 28 ? "#000" :
"#fff" }}} >Month</Text>
            </TouchableOpacity>
            <TouchableOpacity style={{ss.pickerItemContainer, { backgroundColor:
currentInterval === 365 ? "gold" : '#000' }}} onPress={() => onBucketUnitPress(365)}>
                <Text style={{ss.pickerTitle, { color: currentInterval === 365 ? "#000" :
"#fff" }}} >Year</Text>
            </TouchableOpacity>
        </View>
        <View style={{ flex: 1, justifyContent: 'space-between', alignItems: 'stretch', }}>
            {BPMValues.length > 0 && (<View style={{ flex: 1, justifyContent: 'space-
evenly' }}>

```

```

<View>
  <Text style={{ color: '#000', textAlign: 'center' }}>
    <Text style={{ fontWeight: 'bold', fontSize: 30,
}}>{Math.min(...BPMValues)}</Text>
    <Text style={{ fontSize: 25, fontWeight: 'bold' }}>-</Text>
    <Text style={{ fontWeight: 'bold', fontSize: 30,
}}>{Math.max(...BPMValues)}</Text>
    {\t}BPM (Beats Per Minute)
  </Text>
  <Text style={{ marginTop: 50, color: '#000', textAlign: 'center' }}>
    Average:
    <Text style={{ fontSize: 30, fontWeight: 'bold'
}}>{(Math.min(...BPMValues) + Math.max(...BPMValues)) / 2} BPM</Text>
  </Text>
</View>
<View style={{
  width: '100%',
  height: 300,
  backgroundColor: '#fff',
  elevation: 2,
  borderRadius: 10,
  flexDirection: 'row',
  padding: 5
}}>
  <YAxis
    data={BPMValues}
    contentInset={{ top: 0, bottom: 0 }}
    svg={{
      fill: 'grey',
      fontSize: 10,
    }}
    numberOfTicks={10}
    formatLabel={(value) => `${value}`}
  >

```

```

/>
<View style={{ flex: 1, }}>
  <LineChart
    style={{ width: '100%', height: '100%' }}
    data={BPMValues}
    contentInset={{ top: 0, bottom: 0 }}
    svg={{ stroke: "rgba(134, 65, 244, 0.8)", }}
    animate={true}
    withDots={true}
    withHorizontalLabels={true}
    withInnerLines={false}
    withOuterLines={false}
    withVerticalLines={false}
    withHorizontalLines={false}
    numberOfTicks={20}
    curve={shape.curveNatural}
  >
    <Grid direction='HORIZONTAL' />
  </LineChart>
</View>
</View>
</View>)}
{BPMValues.length === 0 && (
  <View style={{ flex: 1, width: '100%', justifyContent: 'center', alignItems:
'center' }}>
    <Text style={{ fontSize: 20, fontWeight: 'bold', color: 'rgba(0, 0, 0, 0.3)'
}}>No data yet.</Text>
  </View>
)}
</View>
</View>
);
}

```

```

export default HeartRateFilterScreen;
const ss = StyleSheet.create({
  screen: { flex: 1, backgroundColor: '#efefef' },
  pickerContainer: {
    width: '100%',
    height: 50,
    backgroundColor: 'grey',
    flexDirection: 'row'
  },
  pickerItemContainer: { justifyContent: 'center', alignItems: 'center', flex: 1,
    backgroundColor: '#000' },
  pickerTitle: { fontSize: 13, fontWeight: 'bold' }
})

```

HeartRateScreen.js

```

import React from 'react';
import { View, Text } from 'react-native';
import { LineChart } from 'react-native-chart-kit';

const sleepData = {
  labels: ["Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"],
  datasets: [
    {
      data: [9, 6, 6.5, 8, 4, 7, 8],
      baseline: 8
    }
  ]
};

const HeartRateScreen = ({
  navigation
}) => {
  return (

```

```

    <View style={{ backgroundColor: '#efefef', justifyContent: 'center', alignItems:
'center', flex: 1 }}>
      </View>
    );
  }
export default HeartRateScreen;

```

Router.js

```

import React from 'react';
import { createStackNavigator } from '@react-navigation/stack'
import AuthroizeScreen from './AuthorizeScreen';
import HeartRateFilterScreen from './HeartRateFilterScreen';
import HeartRateScreenTwo from './HeartRateScreen';
import VitalSelectionScreen from './VitalSelectionScreen';
import StepsVitalScreen from './StepsVitalScreen';

const Stack = createStackNavigator();

const Router = () => {
  return (
    <Stack.Navigator
      screenOptions={{
        headerTitleStyle: { color: '#fff' },
        headerStyle: { backgroundColor: '#000' }
      }}
    >
      <Stack.Screen
        name={"AUTHORIZESCREEN"}
        component={AuthroizeScreen}
        options={{
          headerTitle: "Authorize"
        }}
      />
    </Stack.Navigator>
  );
}

```

```

    <Stack.Screen
      name={"VITALSELECTION"}
      component={VitalSelectionScreen}
      options={{
        headerTitle: "Select Vital",
        headerLeft: null
      }}
    />
    <Stack.Screen
      name={"HEARTRATE"}
      component={HeartRateScreenTwo}
      options={{
        headerTitle: "HeartRates",
        headerLeft: null,
      }}
    />
    <Stack.Screen
      name={"STEPS"}
      component={StepsVitalScreen}
      options={{
        headerTitle: "Track your steps",
        headerLeft: null,
      }}
    />
  </Stack.Navigator>
);
}
export default Router;

```

SleepVitalScreen.js

```

import React from 'react';
import { View, Text, TouchableOpacity } from 'react-native'

```

```

const SleepVitalScreen = ({
  navigation
}) => {
  return (
    <View style={{ flex: 1, backgroundColor: '#efefef', justifyContent: 'center',
    alignItems: 'center' }}>
      <Text style={{ fontSize: 20, }}>No Data to show!</Text>
      <TouchableOpacity onPress={() => navigation.goBack()} style={{ marginTop:
20 }}>
        <Text>Go Back...</Text>
      </TouchableOpacity>
    </View>
  );
}
export default SleepVitalScreen;

```

StepsVitalScreen.js

```

import React, { useEffect, useEffect, useState } from 'react';
import { View, Text, TouchableOpacity, Button } from 'react-native'
import googleFit from 'react-native-google-fit';
import { LineChart, Grid, BarChart } from 'react-native-svg-charts';
import * as shape from 'd3-shape';
import { BucketUnit } from 'react-native-google-fit';

const StepsVitalScreen = ({ navigation }) => {
  const [stepsSample, setStepsSample] = useState([]);
  const [stepsNumberArray, setStepsNumberArray] = useState([]);
  const [interval, setInterval] = useState(7);
  const [startDateFromResponse, setStartDateFromResponse] = useState("");
  const [endDateFromResponse, setEndDateFromResponse] = useState("");
  useEffect(() => {
    navigation.setOptions({
      headerRight: () => (

```

```

<View style={{ flexDirection: 'row' }}>
  <TouchableOpacity
    style={{ height: '100%', justifyContent: 'center', paddingHorizontal: 10,
  }}

    onPress={() => {
      navigation.goBack();
    }}
  >
    <Text style={{ color: '#fff' }}>Back</Text>
  </TouchableOpacity>
  <TouchableOpacity
    style={{ height: '100%', justifyContent: 'center', paddingHorizontal: 10,
  }}

    onPress={() => {
      googleFit.disconnect()
      navigation.replace("AUTHORIZESCREEN")
    }}
  >
    <Text style={{ color: '#fff' }}>Logout</Text>
  </TouchableOpacity>
</View>
)
})
}, [navigation])
useEffect(() => {
  changeInterval(interval);
}, [interval])
useEffect(() => {
  setStepsNumberArray(old => []);
  if(stepsSample.length > 0) {
    let arr = [];
    setStartDateFromResponse(() => stepsSample[0].date)
    setEndDateFromResponse(() => stepsSample[stepsSample.length - 1].date)
  }
}

```

```

        stepsSample.map(item => {
            arr.push(item.value);
        })
        setStepsNumberArray(old => arr);
    }
}, [stepsSample])

const getStepsCount = async (num) => {
    const rightNow = new Date().getTime();
    const endTime = new Date(rightNow);
    const startTime = new Date(rightNow - (num * (24 * 60 * 60 * 1000)));
    console.log(num, ' is the interval');
    console.log(startTime);
    console.log(endTime);
    const stepsData = await googleFit.getDailyStepCountSamples({
        startDate: startTime,
        endDate: endTime,
        bucketInterval: 1,
        bucketUnit: BucketUnit.DAY
    });
    const entimated_steps = stepsData.filter(item => item.source ===
"com.google.android.gms:estimated_steps");
    setStepsSample(old => entimated_steps[0].steps);
}

const changeInterval = (num) => {
    setInterval(old => num);
    getStepsCount(num);
}

return (<View style={{ flex: 1, backgroundColor: '#efefef' }}>
    <View style={{
        width: '100%', height: 50, flexDirection: 'row', backgroundColor: '#000'
    }}>
        <TouchableOpacity

```

```

        onPress={() => changeInterval(1)}

        style={{ flex: 1, justifyContent: 'center', alignItems: 'center', backgroundColor:
interval === 1 ? 'gold' : '#000' }}
    >
        <Text style={{ color: interval === 1 ? '#000' : '#fff', fontWeight: 'bold'
}}>Day</Text>
    </TouchableOpacity>
    <TouchableOpacity
        onPress={() => changeInterval(7)}

        style={{ flex: 1, justifyContent: 'center', alignItems: 'center', backgroundColor:
interval === 7 ? 'gold' : '#000' }}
    >
        <Text style={{ color: interval === 7 ? '#000' : '#fff', fontWeight: 'bold'
}}>Week</Text>
    </TouchableOpacity>
    <TouchableOpacity
        onPress={() => changeInterval(28)}

        style={{ flex: 1, justifyContent: 'center', alignItems: 'center', backgroundColor:
interval === 28 ? 'gold' : '#000' }}
    >
        <Text style={{ color: interval === 28 ? '#000' : '#fff', fontWeight: 'bold'
}}>Month</Text>
    </TouchableOpacity>
    <TouchableOpacity
        onPress={() => changeInterval(365)}

        style={{ flex: 1, justifyContent: 'center', alignItems: 'center', backgroundColor:
interval === 365 ? 'gold' : '#000' }}
    >
        <Text style={{ color: interval === 365 ? '#000' : '#fff', fontWeight: 'bold'
}}>Year</Text>
    </TouchableOpacity>
</View>

{/* If No data is available */}
{

```

```

        stepsSample.length === 0 && (<View style={{ backgroundColor: '#efefef',
justifyContent: 'center', alignItems: 'center', flex: 1 }}>
            <Text style={{ fontSize: 20, fontWeight: 'bold' }}>No data available!</Text>
        </View>)
    }
    { /* If interval is 1 i.e. today */
    {
        stepsSample.length > 0 && interval === 1 && (<View style={{ flex: 1,
backgroundColor: '#efefef', justifyContent: 'center', alignItems: 'center' }}>
            <Text style={{ color: '#000' }}>Last Recorded: </Text>
            <Text style={{ fontSize: 25, color: '#000', fontWeight: 'bold'
}}>{stepsSample[0].date}</Text>
            <Text style={{ fontSize: 15, fontWeight: 'bold', marginTop: 10, color: '#000'
}}>{stepsSample[0].value} Steps</Text>
        </View>)
    }
    {
        (stepsNumberArray.length > 0 && interval !== 1) && (<View style={{ flex: 1,
backgroundColor: '#efefef' }}>
            <View style={{ flex: 1, justifyContent: 'center', alignItems: 'center' }}>
                <Text style={{ fontSize: 15, fontWeight: 'bold', color: '#000' }}>
                    Average:
                    <\t\t'><Text style={{ fontSize: 25
}}>{Math.round((Math.min(...stepsNumberArray) + Math.max(...stepsNumberArray)) /
stepsNumberArray.length)}</Text>
                    <\t'>steps
                </Text>
                <View style={{ marginTop: 20, }}>
                    <Text style={{ fontSize: 20, fontWeight: 'bold', color: '#000' }}><Text
style={{ fontSize: 15, }}>Start Date: </Text>{startDateFromResponse}</Text>
                    <Text style={{ fontSize: 20, fontWeight: 'bold', color: '#000' }}><Text
style={{ fontSize: 15, }}>End Date: </Text>{endDateFromResponse}</Text>
                </View>
            </View>
        <View style={{ flex: 1, justifyContent: 'center', alignItems: 'center' }}>

```

```

    { /* Graph Card */
      <View style={{ width: '90%', height: '70%', borderRadius: 10,
backgroundColor: '#fff', elevation: 3 }}>
        {
          <BarChart style={{ height: 200 }} data={stepsNumberArray} svg={{ fill:
'rgba(0, 0, 0, 0.5)' }} contentInset={{ top: 20, bottom: 20 }}>
            <Grid />
          </BarChart>
        </View>
      </View>
    </View>)
  }
</View>);
}
export default StepsVitalScreen;

```

VitalSelectionScreen.js

```

import React, { useEffect } from 'react';
import { View, Text, TouchableOpacity, StatusBar, FlatList, Dimensions, Image,
ImageBackground } from 'react-native';

const SectionsArr = [
  {
    title: "Heart Rate",
    sectionCode: "HR"
  },
  {
    title: "Track your Steps",
    sectionCode: "STP"
  },
  {
    title: "How's your sleep?",
    sectionCode: "SLP"
  }
]

```

```

]
const VitalSelectionScreen = ({
  navigation
}) => {
  const onPress = (sectionCode) => {
    switch (sectionCode) {
      case "HR":
        navigation.navigate("HEARTRATE")
        return;
      case "STP":
        navigation.navigate("STEPS")
        return;
      case "SLP":
        return;
      default:
        return;
    }
  }

  const renderItem = ({ item, index }) => {
    return (
      <TouchableOpacity style={{ width: '100%', height: 200, marginVertical: 5,
        borderRadius: 10, alignItems: 'center' }} onPress={() => onPress(item.sectionCode)}>
        <ImageBackground
          resizeMode={"center"}
          style={{ backgroundColor: '#fff', width: '95%', height: '100%',
            borderRadius: 10, elevation: 3, justifyContent: 'center', alignItems: 'center' }}
          blurRadius={5}
        >
          <Text style={{ fontSize: 30, fontWeight: 'bold', color: "rgba(0, 0, 0, 1)"
            }}>{item.title}</Text>
        </ImageBackground>
      </TouchableOpacity>);
  }

  return (

```

```

    <View style={{ flex: 1, backgroundColor: '#efefef', justifyContent: 'center',
alignItems: 'center' }}>
      <StatusBar backgroundColor={"#000"} barStyle={"light-content"} />
      <FlatList
        data={SectionsArr}
        renderItem={renderItem}
        contentContainerStyle={{
          width: Dimensions.get('window').width,
        }}
      />
    </View>
  );
}
export default VitalSelectionScreen;

```

GrapUI.js

```

import React, { useEffect, useState } from 'react';
import { View, Text, StyleSheet, Dimensions, ActivityIndicator } from 'react-native';
import { LineChart, Grid } from 'react-native-svg-charts'
import * as shape from 'd3-shape'
import {
  Line
} from 'react-native-svg'
const { width, height } = Dimensions.get('window');
const HeartRateCard = ({
  heartRateSamples
}) => {
  const [heartRateValues, setHeartRateValues] = useState([]);
  useEffect(() => {
    if(heartRateSamples.length > 0) {
      setHeartRateValues(() => [])
      heartRateSamples.map(heartRateObj => {

```

```

        // endDate
        // startDate
        // value
        // day
        setHeartRateValues(old => [...old, heartRateObj.value]);
    })
}
}, [])
return (
  <View style={ss.container}>
    {
      heartRateValues.length > 0 ? (
        <LineChart
          style={{ width: '100%', height: '100%' }}
          data={heartRateValues}
          contentInset={{ top: 0, bottom: 20 }}
          svg={{ stroke: "rgba(134, 65, 244, 0.8)", }}
          animate={true}
          yMin={50}
          yMax={110}
          withDots={true}
          withHorizontalLabels={true}
          withInnerLines={false}
          withOuterLines={false}
          withVerticalLines={false}
          withHorizontalLines={false}
        >
          <Line
            x1={0}
            y1={(width - 150) / 3}
            x2={width - 50}
            y2={(width - 150) / 3}

```

```

        stroke="red"
        strokeWidth="2"
      />
      <Line
        x1={0}
        y1={(width - 150) / 1.2}
        x2={width - 50}
        y2={(width - 150) / 1.2}
        stroke="red"
        strokeWidth="2"
      />
      <Grid svg={null} />
    </LineChart>
  ) : (
    <ActivityIndicator
      color="red"
      size={24}
    />
  )
}
</View>
);
}
export default HeartRateCard;
const ss = StyleSheet.create({
  container: {
    width: width - 50,
    height: width - 250,
    backgroundColor: '#fff',
    borderRadius: 10,
    shadowColor: '#000',
    shadowOffset: { width: 0, height: 1 },

```

```

    shadowOpacity: 0.8,
    shadowRadius: 2,
    elevation: 5,
  },
})

```

Index.js

```

import GoogleFit, {
  Scopes,
} from 'react-native-google-fit';
export const options = {
  scopes: [
    Scopes.FITNESS_ACTIVITY_READ,
    Scopes.FITNESS_ACTIVITY_WRITE,
    Scopes.FITNESS_BODY_READ,
    Scopes.FITNESS_BODY_WRITE,
    Scopes.FITNESS_BLOOD_PRESSURE_READ,
    Scopes.FITNESS_BLOOD_PRESSURE_WRITE,
    Scopes.FITNESS_BLOOD_GLUCOSE_READ,
    Scopes.FITNESS_BLOOD_GLUCOSE_WRITE,
    Scopes.FITNESS_NUTRITION_WRITE,
    Scopes.FITNESS_SLEEP_READ,
    Scopes.FITNESS_HEART_RATE_READ,
    Scopes.FITNESS_HEART_RATE_WRITE,
  ],
};
export const disconnectGFit = () => {
  GoogleFit.disconnect()
}
export const checkIsGFitAuthorized = () => {
  GoogleFit.checkIsAuthorized()
  return GoogleFit.isAuthorized;
}

```

```

export const authorizeGFit = async () => {
  if(GoogleFit.isAuthorized === false) {
    const authResult = await GoogleFit.authorize(options)
    return authResult;
  } else {
    return "Authorized already";
  }
}

export const getHeartRateSamples = async (
  startDate,
  endDate,
  bucketInterval = 15,
  bucketUnit = "MINUTE"
) => {
  if(GoogleFit.isAuthorized === true) {
    const heartRateSmamples = await GoogleFit.getHeartRateSamples({
      startDate,
      endDate,
      bucketInterval,
      bucketUnit
    })
    return heartRateSmamples;
  } else {
    return "You are not authorized.";
  }
}

```

SCREENSHOT



Fig 5.1 – Authorize Screen

This is the screen where the user is being asked to authenticate themselves against google fit

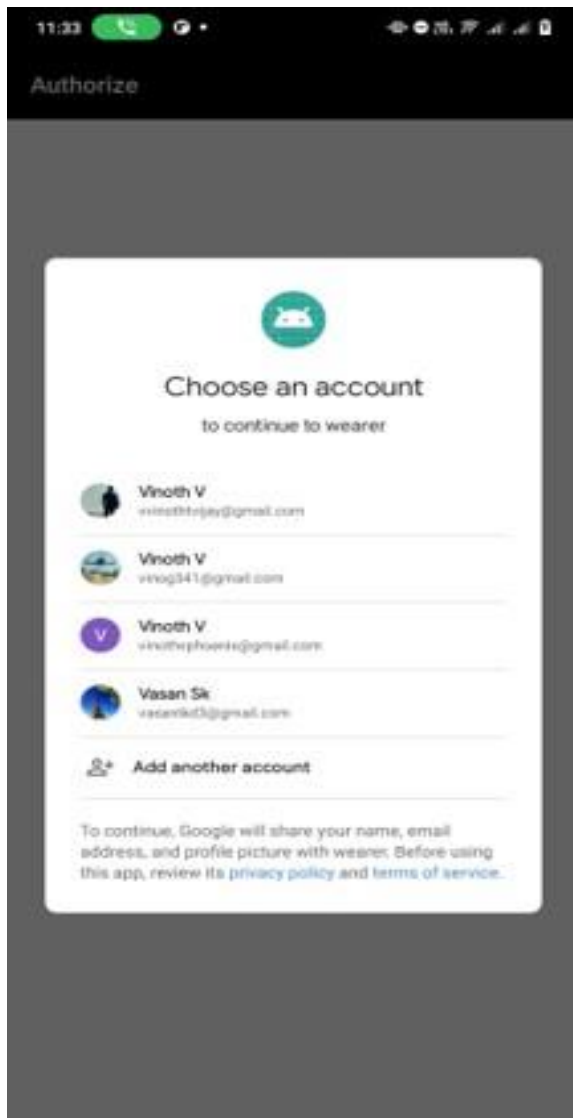


Fig 5.2 – Choosing Account Screen

When user pressed on authorize button shown in the previous screen shot, This module will show the list of Gmails to select one since user will be authenticated against Gmail only.

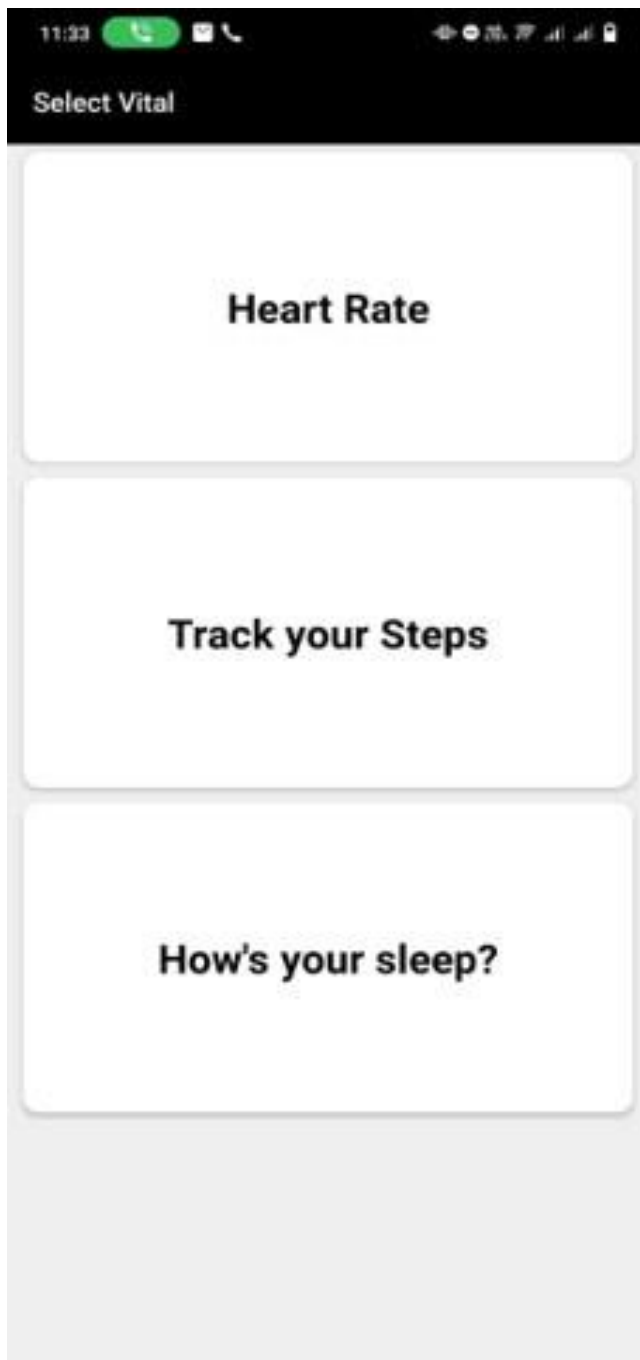


Fig 5.3 – Vitals Screen

After the authentication is successful, it will show list of vitals that this module tracks.

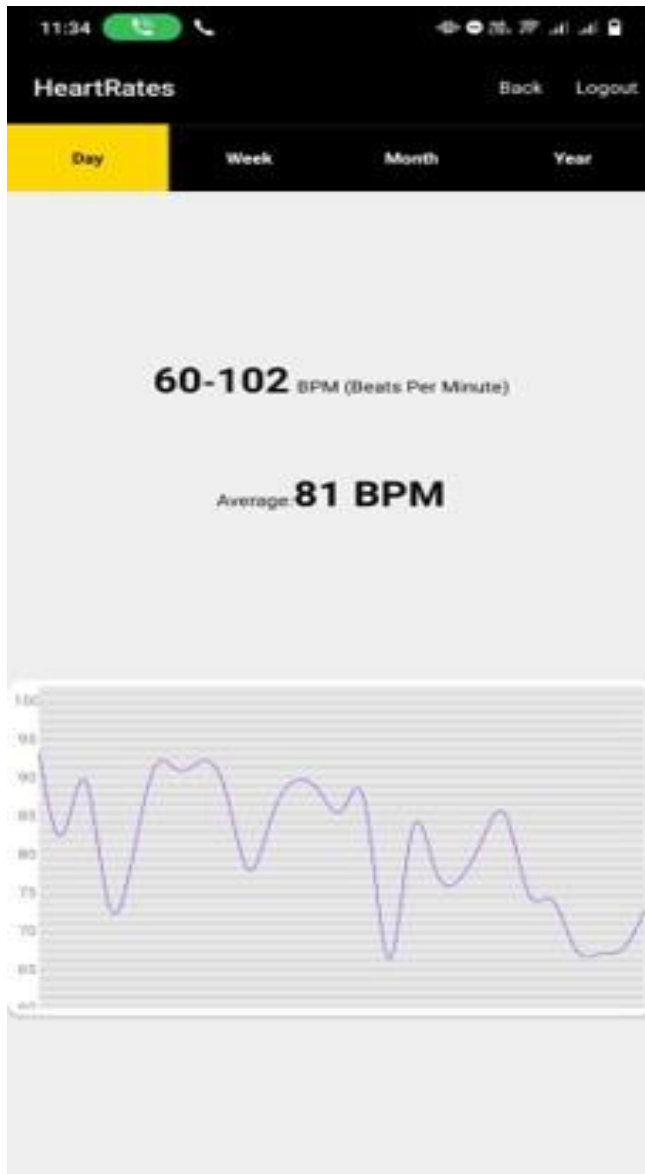


Fig 5.4 – HeartRates Day vice screen

When one vital is selected from the previous screenshot, it will. Show the graph, minimum, maximum and average values of the vitals against provided time range.

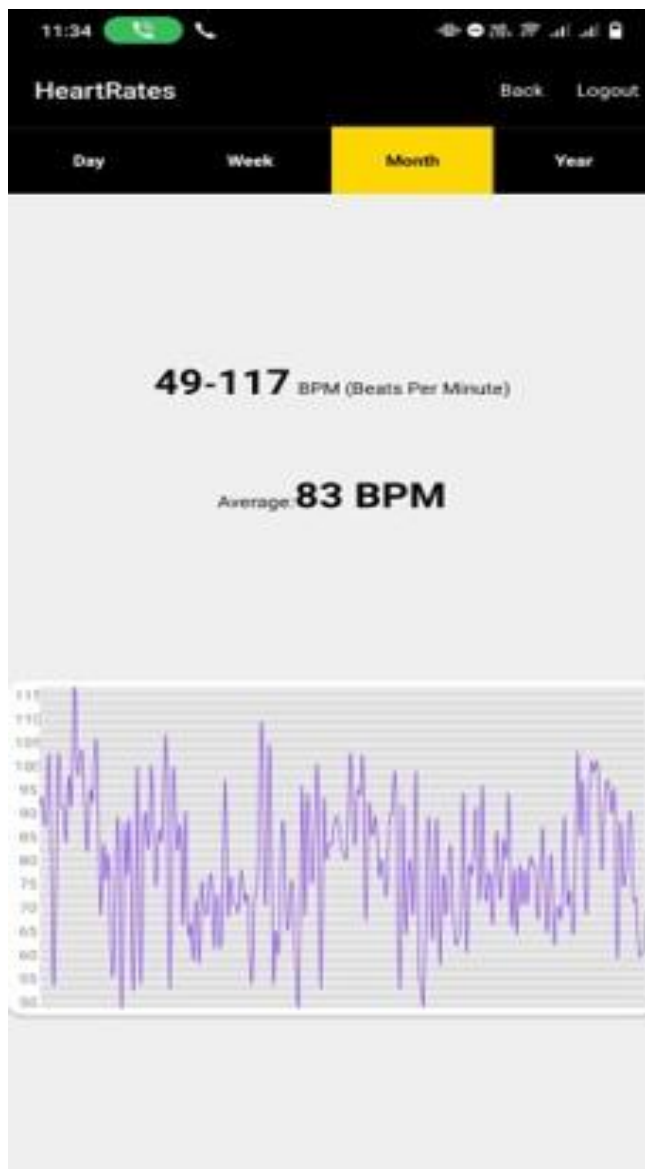


Fig 5.5 – HeartRates Month vice Screen

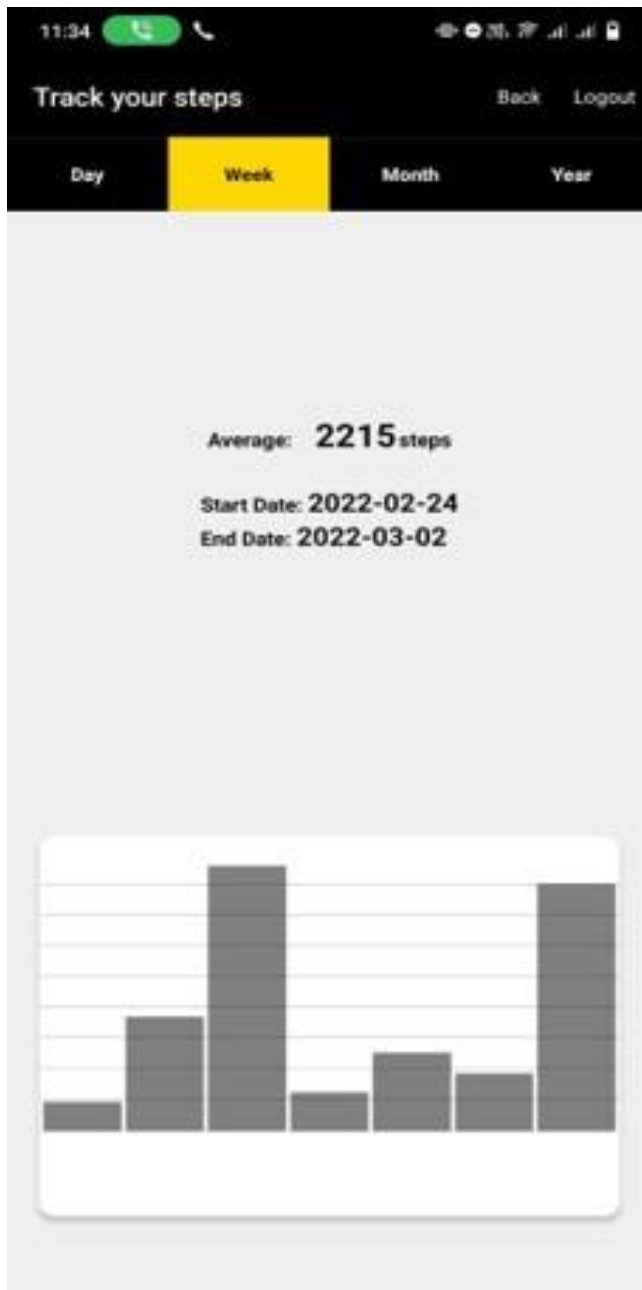


Fig 5.6 – Step Tracking week wise screen

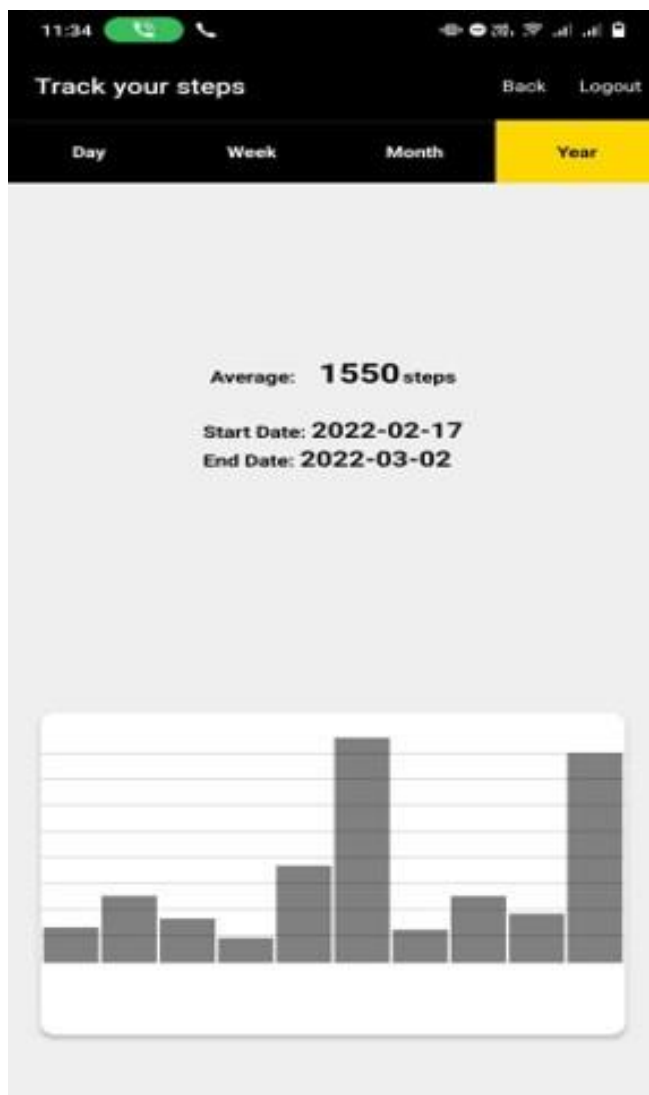


Fig 5.7 - Step Tracking year wise screen