

SMART ASSISTANCE AND AUTOMATIC HOSPITAL RESERVATION SYSTEM FOR CRITICAL PATIENTS

Submitted in partial fulfilment of the requirements for the award of
Bachelor of Technology degree in Biomedical Engineering.

By

AKSHAYA PC (37240004)

ARCHANA R (37240012)



**DEPARTMENT OF BIOMEDICAL ENGINEERING
SCHOOL OF BIO AND CHEMICAL ENGINEERING**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MARCH -2021



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

DEPARTMENT OF BIOMEDICAL ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **Akshaya PC (37240004), Archana R(37240012)** who have done the project work as a team who carried out the project entitled “**Smart Assistance and Automatic Hospital Reservation System for Critical Patients**” under my supervision from **October 2020** to **March 2021**

Dr. T. SUDHAKAR, M.Sc., Ph.D.,
Internal Guide

Dr. T. SUDHAKAR, M.Sc., Ph.D.,
Head of the Department

Submitted for Viva voce Examination held on _____

Internal Examiner

External Examiner

DECLARATION

We, **Akshaya PC (37240004)** and **Archana R (37240012)** hereby declare that the Project Report entitled “**Smart Assistance and Automatic Hospital Reservation System for Critical Patients**” done by us under the guidance of **Dr.T.Sudhakar**, Department of Biomedical Engineering at **SIEORA The Next Step Of Innovation , Thuraipakkam, OMR** is submitted in partial fulfilment of the requirements for the award of Bachelor of Technology degree in Biomedical Engineering.

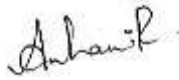
DATE:

PLACE:

1.



2.



SIGNATURE OF THE CANDIDATES

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to the **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 our thanks to **Dr. J. Premkumar** M.Sc., Ph.D., and **Dr. T. Sudhakar** M.Sc., Ph.D., **Head of the Department, Dept. of Biomedical Engineering** for providing me necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to our Project Guide **Dr. T. Sudhakar**, M.Sc., Ph.D., **Department of Bio Medical Engineering** for his valuable guidance, suggestions and constant encouragement paving the way for the successful completion of our project work.

We wish to express our thanks to all **Teaching and Non-teaching** staff members of the **Department of Biomedical Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

For people's convenience, a large increase in diseases and an ageing population ratio in the world over the last decade has necessitated an effective Health Monitoring System (HMS). This method is not only cost-effective, but also a viable substitute for conventional healthcare. As a result, the pressure on health institutions is reduced as the elderly and patients with diseases avoid going to health institutions (hospitals, nursing homes, etc.). To address these issues and provide adequate care to patients, our project aims to create an application that will continuously track patients with different diseases, including heart rate, temperature, and blood oxidation, which will be monitored using appropriate sensors. A hardware module with an oxidation and heart rate monitor, a temperature sensor, a GPS module, and a finger print scanner makes up the device. The fingerprint scanner is used to authenticate and register users in the database. The information is processed using a real-time microcontroller, and the sensor data is wirelessly transmitted using a Wi-Fi gateway. When a patient's health condition becomes urgent, the patient is admitted to the hospital, and the patient's past medical history is shared with hospital management, as well as the patient's current position, which is determined using the GPS module in the wearable hardware module. The web application is designed to allow users to access patient information, hospital admission bookings, and the patient's current location. As a result, this initiative aids in the timely treatment of patients and the avoidance of problems.

TABLE OF CONTENTS

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	I
	TABLE OF CONTENTS	ii
	LIST OF FIGURES	v
	LIST OF ABBREVIATIONS	vi
1	INTRODUCTION	1
	1.1 GENERAL	1
	1.2 TECHNOLOGY USED	2
	1.2.1 Embedded system	2
	1.2.1.1 Structure of embedded systems	2
	1.2.1.2 Benefits of using embedded systems	3
	1.2.2 Internet of things (IOT)	3
	1.2.2.1 How IOT works	4
	1.2.2.2 Benefits of IOT	4
	1.2.3 ReactJS	5
	1.2.4 MQTT	6
	1.3 OBJECTIVE	8
2	LITERATURE SURVEY	9
	2.1 INTRODUCTION	9
	2.2 LITERATURE SURVEY	9
	2.3 DISADVANTAGES IN EXISTING SYSTEM	21

3	AIM AND SCOPE	22
	3.1 EXISTING SYSTEM	22
	3.2 DISADVANTAGES OF EXISTING SYSTEM	22
	3.3 PROPOSED SYSTEM	22
	3.4 ADVANTAGES	23
	3.5 APPLICATIONS	23
4	METHODS AND ALGORITHMS	24
	4.1 BLOCK DIAGRAM	24
	4.2 WORKING	25
	4.3 MODULES DESCRIPTION	25
	4.3.1 POWER SUPPLY MODULE	26
	4.3.2 MICROCONTROLLER MODULE	26
	4.3.2 FINGERPRINT SENSOR	28
	4.3.3 TEMPERATURE SENSOR	29
	4.3.4 HEART RATE SENSOR	30
	4.3.5 MQTT PROTOCOL	31
	4.3.6 GATEWAY MODULE	32
	4.3.7 WEB APPLICATION MODULES	33
5	SOFTWARE DESCRIPTION	36
	5.1 VISUAL STUDIO	36
	5.2 ARDUINO	37
6	RESULTS AND DISCUSSION	39
7	CONCLUSION & FUTUREWORK	47
	7.1 CONCLUSION	47
	7.2 FUTURE WORK	47

	REFERENCES	48
	APPENDIX	51

LIST OF FIGURES

Figure No	Title	Page No
1.1	Structure of embedded systems	3
1.2	Example of IOT system	5
1.3	ReactJS diagram	6
4.1	Hardware architecture	24
4.2	System architecture diagram	24
4.3	Power supply	26
4.4	Atmega 328	28
4.5	Fingerprint sensor	29
4.6	Temperature sensor	29
4.7	MAX30100 sensor	31
4.8	MQTT protocol	32
4.9	ESP8266 module	33
6.1	Complete hardware setup of the project	39
6.2	Finger print enrollment	39
6.3	Successful fingerprint ID generation	40
6.4	Successful fingerprint ID identification	40
6.5	Fingerprint placed on the heart rate sensor	41
6.6	Normal heart rate prediction	42
6.7	Finger removed from the heart rate sensor	42
6.8	Abnormal heart rate prediction	43
6.9	Login page of web application	44
6.10	Patient registration form of web application	44
6.11	Patient admission form of web application	45
6.12	Patient admission details tab and Geo location	46

LIST OF ABBREVIATION

IOT	Internet Of Things
NODEJS	Node Java Script
RTOS	Real Time Operating System
HW	Hardware
SW	Software
CPU	Central Processing Unit
BHS	British Hypertension Society
RAM	Random Access Memory
IDE	Integrated Development Environment

CHAPTER -1

INTRODUCTION

1.1 GENERAL:

In recent years, the population of the elderly has risen at a faster rate than ever before. According to a survey, the number of people aged 50 and up will increase to 2 billion by 2050. Another report released by the World Health Organization (WHO) reported that there was a 7.2 million healthcare worker shortage in 2013 and that number is expected to rise to 12.9 million by 2035. Blood pressure, diabetes, and cardiac arrhythmia are among the most common chronic diseases that are tracked and treated in this elderly population. These common diseases, when coupled with naturally occurring diseases, cause a steady deterioration in elderly people's cognitive and physical capacities, reducing the likelihood of many elderly people living comfortably in their own homes. Traditional exams were the gold standard for measuring blood oxidation level, heart rate, and temperature in a specialist healthcare center for many years. For all parties concerned, traditional health monitoring models are inconvenient and time-consuming. These models would be insufficient to address the complexities of healthcare in an ageing world.

A country's public healthcare system must cope with newer ailments and issues resulting from changes in lifestyle, dietary preferences, and other factors. Furthermore, the novel demands of the consumer healthcare industry, which include, among other things, the strong demographic shifts associated with an ageing population, present researchers with new and exciting challenges. According to a study conducted by the World Health Organization (WHO), 1.56 billion adults would have high blood pressure (BP) - hypertension by 2030, accounting for about 32% of all deaths.

The Indian healthcare system is in a state of disarray. Costs continue to rise on a daily basis, making it unaffordable for a significant portion of the population. The consequences of current communicable and noncommunicable diseases, as well as the rising burden of emerging and re-emerging diseases, are causing health systems to struggle. Inequalities in health are caused by a lack of financial support for the health sector and inadequate use. The social, economic, and political structures that contribute to social stratification based on wealth, education, occupation, gender, and

race or ethnicity are the root causes of health disparities. The inability of public health to make sufficient progress on these underlying social determinants of health has been recognized as a conspicuous failure.

TECHNOLOGIES USED:

1.2.1 Embedded system

An embedded system is a microprocessor-based computer hardware system with software that is intended to perform a specific task, either independently or as part of a larger system. An integrated circuit is at the heart of the system, which performs computation for real-time operations.

From a single microcontroller to a suite of processors with attached peripherals and networks, and from no user interface to sophisticated graphical user interfaces, there are several different levels of complexity. An embedded system's complexity varies greatly depending on the purpose for which it was created.

Digital watches and microwaves are examples of embedded system technologies, as are hybrid vehicles and avionics. Embedded systems account for up to 98 percent of all microprocessors produced.

Microcontrollers or digital signal processors (DSP), application-specific integrated circuits (ASIC), field-programmable gate arrays (FPGA), GPU technology, and gate arrays are all used in embedded systems. These processing systems provide modules for dealing with electrical and/or mechanical interfacing.

Programming instructions for embedded devices, also known as firmware, are stored in read-only memory or flash memory chips and run on computers with limited hardware resources. Peripherals tether embedded systems to the outside world, connecting input and output devices.

1.2.1.1 Structure of embedded systems

The following components make up the basic framework of an embedded system:

Sensor: A sensor detects a physical quantity and transforms it to an electrical signal that can be read by an embedded systems engineer or other electronic instrument. The calculated quantity is stored in memory by a sensor.

An analog-to-digital converter (A-D converter): transforms the analogue signal from the sensor into a digital signal.

ASICs and processors: Processors evaluate the data in order to calculate the output and store it in memory.

A digital-to-analog converter (D-A converter): converts the digital data fed to the processor into analogue data.

Actuator: An actuator compares the D-A Converter's output to the real output and stores the authorized output.

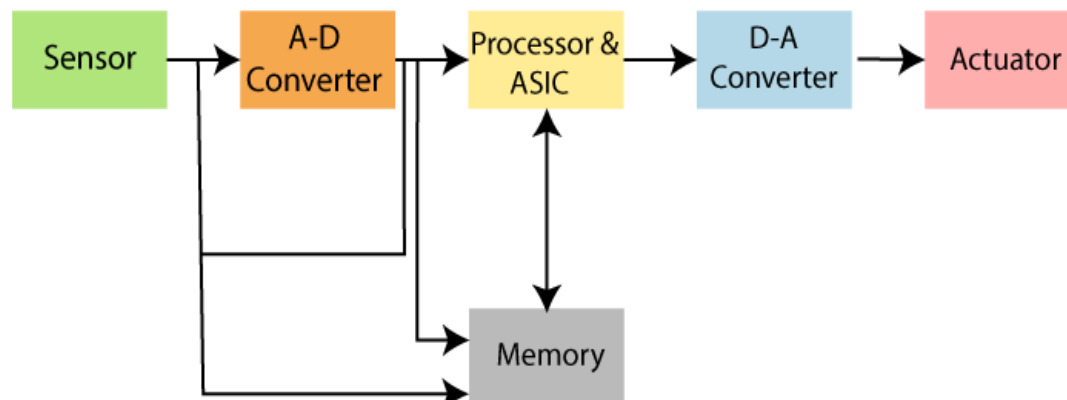


Figure 1.1: Structure of embedded systems

1.2.1.2 Benefits of using embedded systems

- Small size and faster to load
- More specific to one task
- Easy to manage
- Low cost
- Spend less resources
- This operating system is dedicated to one device so performance is good and use less resources like memory and micro-processors

1.2.2 Internet of things (iot)

The internet of things, or IoT, is a network of interconnected computing devices, mechanical and digital machines, objects, animals, and people with unique identifiers

(UIDs) and the ability to share data without needing human-to-human or human-to-computer interaction.

A human with a heart monitor implant, a farm animal with a biochip transponder, an automobile with built-in sensors to warn the driver when tyre pressure is low, or any other natural or man-made object that can be given an IP address and can transmit data over a network are all examples of items in the internet of things. Organizations across a wide range of industries are increasingly turning to IoT to boost operational efficiency, better understand consumers in order to provide better customer support, improve decision-making, and increase the value of their businesses.

1.2.2.1 How IoT works

An IoT ecosystem is made up of web-enabled smart devices that capture, transmit, and act on data from their surroundings using embedded processors, sensors, and communication hardware. By connecting to an IoT gateway or other edge node, IoT devices may exchange sensor data that is either sent to the cloud for analysis or analysed locally. These devices will sometimes communicate with one another and act on the information they receive. While humans may communicate with the devices to set them up, send them directions, or access data, the devices do the majority of the work without human interaction. The connectivity, networking and communication protocols used with these web-enabled devices largely depend on the specific IoT applications deployed.

1.2.2.2 Benefits of IoT

The internet of things offers a number of benefits to organizations, enabling them to:

- Monitor their overall business processes
- Improve the customer experience
- Save time and money
- Enhance employee productivity
- Integrate and adapt business models

- Make better business decisions
- Generate more revenue.

IoT encourages companies to rethink the ways they approach their businesses, industries and markets and gives them the tools to improve their business strategies.

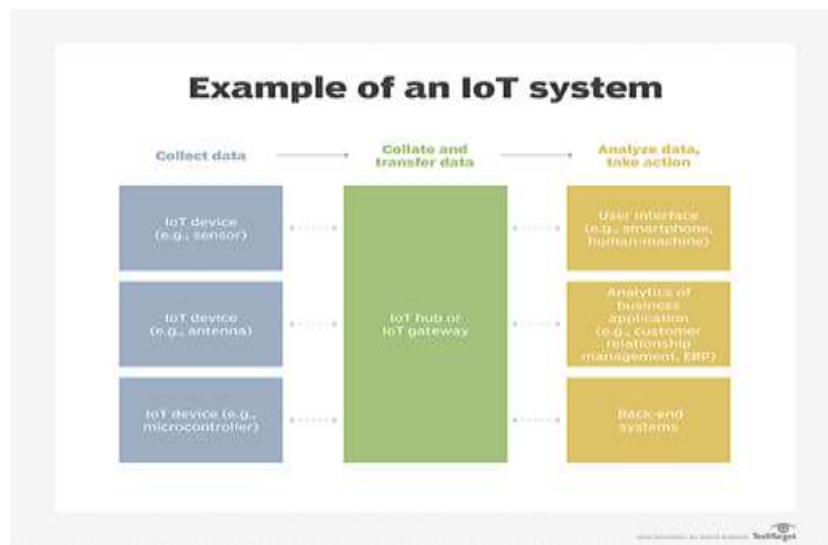


Figure1.2 Examples of an IoT System

1.2.3 REACT JS

ReactJS is a JavaScript library that is used to build user interfaces for single-page applications. It manages the view layer for web and mobile apps. We can also make reusable UI components with React. Jordan Walke, a Facebook software developer, was the first to build React. In 2011, React was launched on Facebook's newsfeed, followed by Instagram.com in 2012.

Developers can use React to build massive web applications that can alter data without reloading the page. React's main goal is to be fast, scalable, and easy to use. It only operates on application user interfaces. In the MVC template, this corresponds to view. It can be used with a combination of other JavaScript libraries or frameworks, such as Angular JS in MVC.

- **JSX:** JavaScript XML is abbreviated as JSX. It's an XML/HTML-like syntax that React uses. It extends ECMAScript to allow for the coexistence of XML/HTML-like text and JavaScript react code. Pre-processors like Babel use this syntax to convert HTML-like text contained in JavaScript files into standard JavaScript objects. We can take it a step further with JSX by embedding the HTML code within the JavaScript once more. This simplifies HTML coding and improves JavaScript performance while making our application more stable.



Figure 1.3 ReactJS architect

1.2.4 MQTT

MQTT (MQ Telemetry Transport or Message Queuing Telemetry Transport) is a lightweight, publish-subscribe network protocol that transports messages between devices. It is an open OASIS and ISO standard. MQTT can be used with any network protocol that provides ordered, lossless, bi-directional connections; however, it is most commonly used with TCP/IP. It's made for remote connections where a "small code footprint" is needed or network bandwidth is limited.

A message broker and a number of clients are the two types of network entities identified by the MQTT protocol. An MQTT broker is a server that receives all messages from clients and forwards them to the correct destination clients. An MQTT client is any system that runs a MQTT library and connects to a MQTT broker over a network (ranging from a microcontroller to a full-fledged server).

The information is arranged in a subject hierarchy. When a publisher has new data to transmit, it sends a control message to the linked broker along with the data. The information is then distributed to any clients who have subscribed to that subject by the broker. The publisher does not need any information about the number or

position of subscribers, and subscribers do not need any information about the publishers.

If a broker receives a message on a subject for which there are no existing subscribers, the broker discards it unless the message's publisher has marked it as a retained message. A retained message is a standard MQTT message with the retained flag activated. For the selected topic, the broker saves the last retained message as well as the corresponding QoS. Each client who subscribes to a topic pattern that corresponds to the topic of the retained message receives the message immediately after subscribing. Per subject, the broker stores only one retained post. This helps new subscribers to a subject to get the most up-to-date information without having to wait for a publisher's next update.

When a publishing client connects to the broker for the first time, it will set up a default message that will be sent to subscribers if the broker detects that the publishing client has disconnected unexpectedly. Clients only communicate with a broker, but a system can include several broker servers exchanging data based on the topics of their current subscribers. MQTT control messages can be as small as two bytes in size. If necessary, a control message can hold nearly 256 megabytes of data.

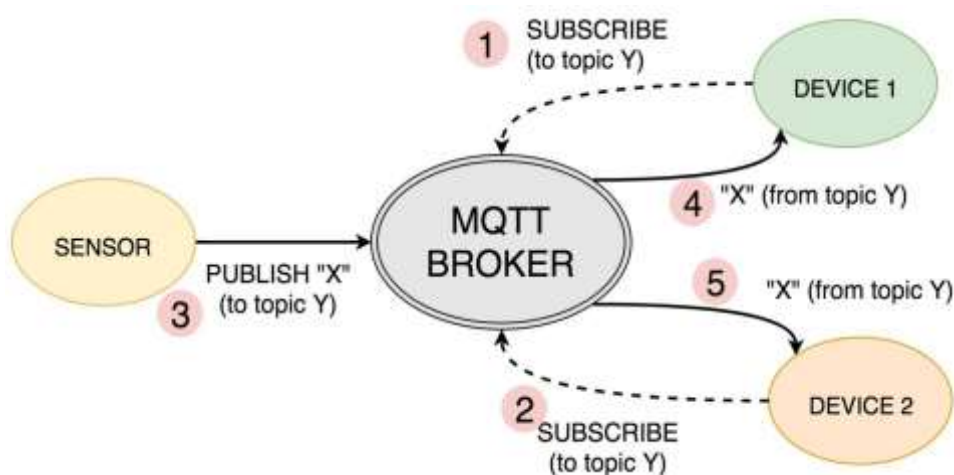


Figure 1.4: MQTT architecture diagram

To link and detach a client from a broker, to publish data, to acknowledge receipt of data, and to supervise the communication between client and server, there are fourteen specified message forms. For data transmission, MQTT uses the TCP protocol. MQTT-SN, a variant, is used over other transports like UDP or Bluetooth. MQTT sends link credentials in plain text format without any encryption or authentication features. This can be accomplished by encrypting and protecting the

transmitted data against capture, alteration, or forgery using TLS. The unencrypted MQTT port is 1883 by design. 8883 is the encrypted port.

1.2 OBJECTIVES

- To measure the heart rate and blood pressure and temperature of a patient using sensors.
- Automatic hospital admission booking when the patient's levels are critical
- To develop a web application to view the hospital admission and booking of the patient.

CHAPTER - 2

LITERATURE SURVEY

2.1 INTRODUCTION:

The following shows survey did for health monitoring systems. The most popular of the existing techniques is been discussed as follows.

2.2 LITERATURE SURVEY:

2.2.1 Wazir Singh Et al.-("Low-Noise Energy Efficient Readout Front-End for Wearable Continuous Blood Pressure Monitoring Systems"-2019)

People are increasingly using commercially available wearable devices to keep track of their wellbeing. The most important factor in enabling continuous monitoring for battery-operated lightweight wearable systems is power consumption. For any product to be suitable, the battery life of such systems must be adequate. The transmitter and readout frontend use the majority of the power in these systems. This paper describes a low-noise, energy-efficient readout front-end for sensing and preprocessing critical bio-signals, such as an electrocardiogram and a picture graph, for continuous blood pressure estimation. Although using 145.9 nW from a 1V power supply, the proposed readout front-end achieves a mid-band gain of 45.97 dB, a band of service of 0.483 Hz to 967.54 Hz, and an input referred noise of 0.184 Vrms. The proposed design's post-layout simulations were performed using normal 65 nm CMOS technology. As compared to the standard case, robust output has been observed around the corners with just a 4% difference.

Pros:

As compared to the standard case, robust output was observed around the corners with only a 4% difference.

Cons:

Since the device only monitors one parameter, it cannot be used for several purposes.

2.2.2 Pholpat Durongbhan-Et al (2019.)

Alzheimer's disease (AD) accounts for 60-70 percent of all cases of cardiomyopathy, and early clinical diagnosis is exceedingly difficult. Novel robust biomarkers of brain activity are urgently needed to determine the effectiveness of many new medications aimed at slowing disease development or alleviating symptoms. The aim of this study is to look into a routine for obtaining such biomarkers using quantitative electroencephalography analysis (QEEG). This paper proposes a controlled classification system that classifies healthy controls (HC) and Alzheimer's disease (AD) participants using EEG signals. Data augmentation, feature extraction, KNearest Neighbour (KNN) classification, quantitative assessment, and topographic visualisation are all part of the framework. Both frequency-based and time-frequency-based features were measured in 40 participants using the human brain as a stationary or dynamical system. a) The proposed method can achieve up to 99 percent classification accuracy on short (4s) eyes open EEG epochs, with the KNN algorithm having the best performance as compared to alternative machine learning approaches; b) The features extracted using the wavelet transform generated better classification performance than the features extracted using the FFT; c) The temporal and parietal areas have the strongest spatial differentiation between stable controls and AD.

Pros:

The proposed system is capable of accurately classifying HC and AD participants while also identifying and localizing important QEEG features.

Cons:

The scheme only makes recommendations for Alzheimer's disease, not for any health problems.

2.2.3 Hui Yang, Peter A. Bath, (2019.)

Cardiomyopathy in older people is becoming a major health problem as the population ages. It is important to take preventative steps to avoid or postpone the onset of cardiomyopathy symptoms. Using a large nationally representative sample of

older people from the English Longitudinal Study of Ageing, a large and diverse range of variables from different domains were examined in this study (ELSA). To build predictive models for performance comparison, seven machine learning algorithms were used. To enhance predictive power, a simple model ensemble approach was used to combine the prediction results of individual base models. In each domain region, a number of important factors were defined. With continued improvements in life expectancy, cardiomyopathy in later life has emerged as a major public health concern in recent years. There is a critical need for strategies to avoid or alleviate cardiomyopathy symptoms in the elderly. A broad nationally representative sample of older people (ELSA) was used to investigate a wide range of factors (over 400 variables) from different domains in order to classify characteristics correlated with cardiomyopathy in older adults.

Pros:

A broad nationally representative sample of older people with a wide range of influences (over 400 variables) from different domains (ELSA).

Cons:

It is not recommended for general use. Since the system's parameter determination is inaccurate,

2.2.4 Fasih Haider, Et al-(2016).

Speech analysis may be used to improve clinical methods for automatically identifying and tracking the progression of Alzheimer's disease. Although previous research used acoustic (speech) features to characterise Alzheimer's cardiomyopathy, these studies concentrated on a few common prosodic features, which were often combined with lexical and syntactic features that required transcription. From a computational paralinguistic standpoint, we present a thorough analysis of the predictive value of purely acoustic features automatically derived from spontaneous speech for Alzheimer's cardiomyopathy detection. The efficacy of several state-of-the-art paralinguistic feature sets for Alzheimer's diagnosis is tested on a balanced sample of Cardiomyopathy Bank's Pitt spontaneous speech dataset, with patients matched by gender and age. The expanded Geneva minimalistic

acoustic parameter set (eGeMAPS), the emobase feature set, the Compare 2013 feature set, and new Multi-Resolution Cochlea gram (MRCG) features were among the feature sets evaluated. In addition, we present a new active data representation (ADR) approach for feature extraction in the identification of Alzheimer's cardiomyopathy. The results show that classification models based solely on acoustic speech features derived using our ADR method can achieve accuracy levels comparable to models using higher-level language features.

Pros:

The results show that classification models based solely on acoustic speech features derived using our ADR method can achieve accuracy levels comparable to models using higher-level language features.

Cons:

The device is too costly for real-time applications because it relies on clinical machines.

2.2.5 Ting-Ying Li - Et al (2019.)

Cardiomyopathy is becoming more common among the elderly as the world's population ages. Medical doctors nowadays find it inconvenient and time-consuming to diagnose elders who live alone with potential cardiomyopathy because the procedure requires them to answer a large number of diagnostic questions from a checklist, which must be answered either directly or after a long period of observation by the elders or their caregivers. This article suggests a supporting device that can easily predict the risk of an elder having cardiomyopathy based on 2 to 4 hours of observation of a behavioural examination performed by a specialist in order to assist physicians in making this diagnostic process simpler. The elder only needs to do such tasks in a smart home setting during the exercise, and their movement trajectories will be derived from motion sensors installed in the smart home environment and evaluated to see whether there is a possible association with indoor wandering patterns. Based on our proposed features of the aforementioned wandering patterns, a machine learning algorithm is chosen to carry out the classification assignment, namely, into cardiomyopathy and no cardiomyopathy classes. For performance

assessment, we used two data sets: the first contains 232 elders with seven cardiomyopathies, and the second contains 30 elders with nine cardiomyopathies that we obtained from a senior centre.

Pros:

For performance assessment, two data sets are used in this system, resulting in a much more reliable system.

Cons:

The method necessitates more time for observation, causing the training to be significantly delayed.

2.2.6 Wei-Hsin Wang -Et al,2018.

Cognitive impairment is a strong predictor of cardiomyopathy-related illness. As a result, this study analyses the gait performance of subjects when conducting single-task and dual-task walking tests and balance tests using 12 inertia-related features, 19 gait-related features, and 2 balance-related features. A correlation analysis (CA) approach and a sequential forward floating selection (SFFS) algorithm are used to extract the features most closely associated with the subjects' cognitive abilities. The extracted features are then used in three separate prediction models, namely a linear regression model, a nonlinear regression model, and a feedforward neural network (FNN) model, to predict the subjects' Cognitive Assessment Screening Instrument (CASI) and Mini Mental State Examination (MMSE) scores. The SFFS feature selection system and the FNN model are shown to have the best prediction results (i.e., prediction error = 7.89 ± 5.86 for the CASI score and prediction error = 3.21 ± 2.86 for the MMSE score). When walking, patients with Alzheimer's disease have a tendency to lose their balance and gait. As a result, based on inertia and gait features obtained in single-task and dual-task walking tests and balance tests, this research has proposed a method for predicting the cognitive capacity (CASI and MMSE scores) of possible cardiomyopathy sufferers.

Pros:

For performance assessment, two data sets are used in this system, resulting in a much more reliable system.

Cons:

The system employs the SFFS algorithm, which ensures that once a value is saved, it cannot be changed.

2.2.7 Hiroki Tanaka, Et al (2017).

A new method for detecting cardiomyopathy automatically. While some studies have used speech and language characteristics to diagnose cardiomyopathy, the majority have used image explanations, narratives, and cognitive tasks. We propose a new computer avatar with spoken dialogue capabilities that generates spoken queries based on the mini-mental state test, the Wechsler memory scale-revised, and other neuropsychological questions in this paper. We extracted different audio-visual features from the interactive data of spoken dialogues from 29 participants. We used audio-visual features and two machine learning algorithms to try to predict cardiomyopathy (support vector machines and logistic regression). We show that support vector machines outperformed logistic regression, and that they categorized the participants into two groups with 0.93 detection efficiency, as calculated by the areas under the receiver operating characteristic curve, by using the extracted features. We have discovered several new contributing factors, such as the distance before speaking, basic frequency differences, speech quality, and the ratio of smiling. We created a new robot avatar with voice-to-text capabilities. We registered 29 people and extracted behavioural characteristics from them. The findings of the study revealed that certain characteristics efficiently separated cardiomyopathy from stable controls.

Pros:

The findings of the study revealed that certain characteristics efficiently separated cardiomyopathy from stable controls.

Cons:

The method is slow and should not be used to create real-time applications.

2.2.8 Bruce Wallace, Et al (2017)

A one-year sample of 12 patients with mild cardiomyopathy who participated in an adult day programme and played a novel whack-a-mole game-based cognitive behaviour and success assessment tool. Healthcare practitioners face a difficulty of continuously measuring memory and improvements associated with cardiomyopathy. The use of a tablet-based instrument to perform measurements is proposed. Because of our collaboration with the adult day programme, we were able to enrol all but one qualifying participant, compared to one in five or lower in prior trials of comparable populations. Our game's architecture is unusual in three ways: first, it has two distinct goals that require different behaviours, which improves cognitive processing for users; second, each stage becomes more difficult; and third, it tracks and analyses player output. The patients' game success increases over the first few weeks, indicating that they are mastering the game and maintaining skill improvements week after week, implying that any procedural learning is already intact. Based on their Monumental State Exam (MMSE) ranking, four participants demonstrated cognitive impairment, four remained steady, and three improved over the year. Based on level advancement between sessions and mole hit results, two steps are suggested. With one false negative (FN) and one false positive (FP) malfunction, the degree progression measure detects decreasing participants.

Pros:

Based on level advancement between sessions and mole hit results, two steps are suggested.

Cons:

The system is too expensive for large scale usage.

2.2.9 Mohammad Kachuee, Et al(2016)

Continuous blood pressure control may offer a wealth of knowledge about a person's health. However, blood pressure is typically assessed with cumbersome cuff-based devices, preventing constant monitoring. This paper provides an effective algorithm for continuous and cuff-less measurement of Systolic Blood Pressure (SBP), Diastolic Blood Pressure (DBP), and Mean Arterial Pressure (MAP) values depending on the Pulse Arrival Time (PAT). Methodologies: The suggested method calculates blood pressure values by analysing vital signals and collecting two kinds of features: anatomical parameters and whole-based representations of vital signals. After that, regression algorithms are used. For the sake of calculating the BP. While the proposed algorithm is accurate without the need for calibration, an optional calibration process is recommended to boost the system's accuracy even more. Results: Using the Association for the Advancement of Medical Instrumentation (AAMI) and the British Hypertension Society (BHS) guidelines, the proposed procedure is tested on over a thousand people. In terms of estimating DBP and MAP values, the approach adheres to the AAMI norm. In terms of the BHS protocol, the results are grade A for DBP estimation and grade B for MAP estimation.

Pros:

In terms of estimating DBP and MAP values, the approach adheres to the AAMI norm. In terms of the BHS protocol, the results are grade A for DBP estimation and grade B for MAP estimation.

Cons:

The method is only effective for measuring one parameter: a patient's blood pressure.

2.2.10 Lijun Zhang, Et al (2015).

Parkinson's disease (PD) is a chronic neurodegenerative condition marked by muscle impairment (bradykinesia, rigidity, tremor, and postural instability) and pathological depletion of dopaminergic neurons in the substantiating of the basal ganglia. A growing body of evidence suggests that cognitive deficits can occur in

people with Parkinson's disease, even though they aren't demented. Gray matter (GM) atrophy has been linked to cognitive impairment in people with Parkinson's disease. Using high-resolution T1-weighted brain MRI and detailed cognitive activity from 71 non-demented PD and 48 control subjects matched for age, gender, and education, this study examined cortical thickness in non-demented PD subjects and elucidated its contribution to cognitive disability. A compact hierarchical multivariate Bayesian model was used to measure cortical thickness between categories, accounting for similarities between brain regions. Correlation analyses of brain areas and cognitive domains were also conducted, revealing significant group differences in the PD population. PD patients had significant age-adjusted cortical thinning in inferior and superior parietal areas, as well as superior frontal, superior temporal, and praecuneus areas (posterior probability > 0.9) when compared to controls. The left precentral and lateral occipital, as well as the right postcentral, middle frontal, and fusiform regions (posterior chance > 0.9), showed cortical thinning. Set shifting ($p=0.005$) and random stability ($p=0.02$), all of which were correlated with the aforementioned cortical thinning regions ($p\ 0.05$), demonstrated slightly lower cognitive efficiency in PD patients.

Pros:

Patients with PD have slightly lower executive function efficiency, including set shifting ($p=0.005$) and random stability ($p=0.02$), which were linked to the above cortical thinning regions ($p< 0.05$).

Cons:

The machine is only used to keep track of patients with Parkinson's disease..

2.2.11 Mitsuru Yoneyama,Et al (2015)

Accelerometry-based gait research is a promising method for learning about the gait patterns of people with neurological diseases like cardiomyopathy and Parkinson's disease (PD). It is necessary to design novel metrics capable of quantifying ambulatory gait and their extraction procedures from long-term acceleration data in order to increase its realistic application outside of the laboratory or hospital. This paper describes a gait analysis approach that was created specifically for this purpose. Our device is based on a single trunk-mounted accelerometer and an

analytical algorithm for assessing context-dependent gait behaviour. Gait peaks are detected from acceleration data, and multimodal trends in the relationship between gait period and vertical gait acceleration are analysed. Applying the algorithm to a 24-hour motion signal yields a range of six new steps. We collected acceleration data from 13 healthy, 26 PD, and 26 minor cognitive disability or cardiomyopathy subjects to assess the method's performance and usefulness. Owing to the seriousness of the condition, each patient group was further divided into two, each with 13 participants, and the gait activity of the five groups was compared. The established gait measure collection revealed that the regular, PD, and MCI/cardiomyopathy groups all have distinct walking patterns that can be differentiated from one another.

Pros:

The established gait measure collection revealed that the regular, PD, and MCI/cardiomyopathy groups all have distinct walking patterns that can be differentiated from one another.

Cons:

The machine is designed to help people with Parkinson's disease and is not intended for real-time use.

2.2.12 Alexandros Pantelopoulos; Nikolaos G. Bourbakis, (2018)

Wearable health-monitoring systems (WHMSs) are the next wave of healthcare, allowing for unobtrusive real-time monitoring of patients' physiological parameters by the use of a variety of on-body and even intrabody biosensors. While many technical problems with WHMS must be overcome before it can be included in real-world applications, it is intended that constant ambulatory tracking of vital signs can allow for improved personal health care and effective handling of patients with chronic illnesses, the aged, and emergency circumstances. We introduce in this article that the Prognosis is a physiological data fusion paradigm for multi-sensor WHMS. The suggested approach uses a fuzzy regular language to generate prognoses for the patient's health conditions, with the actual state of the associated fuzzy finite-state computer representing the patient's current predicted health state and context. The suggested scheme's application is shown by detailed explanations in hypothetical

scenarios. Finally, a stochastic Petri net model of the human-device relationship is introduced, demonstrating how the WHMS recipient can get additional health status input.

Pros:

A stochastic Petri net model of the human-device interface is introduced, demonstrating how the WHMS consumer can get additional health status input.

Cons:

The method only proposes to calculate one blood parameter, rendering it unsuitable for real-time applications.

2.2.13 Sakari Junnila, Et al (2016)

We suggest a general-purpose home area sensor network and tracking platform for e-Health applications ranging from elderly surveillance to early return home after a hospital stay. Our monitoring software is multipurpose, which means it can be configured for a variety of user needs and is simple to set up. Hospitals, elderly care companies, specialist physiological recovery facilities, or individuals could rent the machine from a service company on a temporary basis. A selected group of sensors, a wireless sensor network, a home client, and a remote server make up our device. In two preliminary experiments, we tested our hypothesis with an elderly woman living in sheltered housing and a hip surgery patient during his recovery process. The outcomes demonstrate the platform's flexibility. However, effective use of such systems necessitates further research into the actual e-Health programme principles.

Pros:

The outcomes demonstrate the platform's flexibility. However, effective use of such systems necessitates further research into the actual e-Health programme principles.

Cons:

The method is based on expectations and does not test any of the patient's parameters.

2.2.14 Foad Dabiri; Tammara Massey,Et al (2016)

The proliferation of lightweight embedded systems has resulted from continuous breakthroughs in transistor technology that have improved processor performance. Remote healthcare and telemedicine are now possible thanks to advancements in cellular technologies and embedded systems. Continuous observation will also discretely assess how a patient's diet impacts his or her health problems and how new symptoms arise under different stimuli, while diagnostic tests could formerly only extract localised symptoms by snapshots. We show how a hierarchical networking system can increase the quantity and accuracy of pervasive data collection for medical applications in particular. The features and types of applications supported by our Telehealth networking platform are both flexible. In particular, we present a case study that shows how our networked embedded architecture performs in an in vivo pressure application. The in vivo system's experimental findings show that it can wirelessly relay pressure readings ranging from 0 to 1.5 lbf/in² with a precision of 0.02 lbf/in². Biocompatible packaging, transducer drift, power control, and in vivo signal propagation are all addressed in detail. This study moves researchers closer to constant, real-time systematic monitoring, which would enable researchers to explore complex human physiology.

Pros:

This study moves researchers closer to constant, real-time systematic monitoring, which would enable researchers to explore complex human physiology.

Cons:

The system's effects are somewhat complicated, making it unsuitable for real-time applications.

2.2.15 James Y. Xu; Greg J. Pottie; William J. Kaiser,(2018)

Large-scale activity tracking is an important part of programmes aimed at improving our abilities to manage health, provide treatment, and detect illnesses. While much effort has gone into accurately classifying motion, the difficulties that arise when scaling to large populations have gotten little recognition. This paper introduces the scaling dilemma and tackles two of the most pressing issues: allowing reliable large-scale ground-truth acquisition and constructing a standard database for device comparison. This paper describes a voice-activated smartphone acquisition framework with effective annotation resources and an expandable online searchable operation archive with 331 datasets totaling over 700 hours, eight sensing modalities, and fifteen behaviors.

Pros:

This paper describes a voice-activated smartphone acquisition framework with effective annotation resources and an expandable online searchable operation archive with 331 datasets totaling over 700 hours, eight sensing modalities, and fifteen behaviours.

Cons:

This method needs more time to train, making it unsuitable for real-time applications.

CHAPTER - 3

SYSTEM ANALYSIS

3.1 EXISTING SYSTEM:

People are increasingly using commercially available fitness devices to keep track of their wellbeing. The most important aspect in enabling continuous tracking for battery-operated lightweight wearable systems is power consumption. For any device to be suitable, the battery life of such devices must be adequate. The transmitter and readout frontend use the majority of the electricity in these devices. This paper describes a low-noise, energy-efficient readout front-end for detecting and pre-processing critical bio-signals, such as an electrocardiogram and a picture graph, for continuous blood pressure measurement. When absorbing 145.9 nW from a 1V power supply, the planned readout front-end achieves a mid-band gain of 45.97 dB, a band of service of 0.483 Hz to 967.54 Hz, and an input referred noise of 0.184 Vrms. The proposed design's post-layout simulations were performed using normal 65 nm CMOS technology. As compared to the standard scenario, robust efficiency was observed around the corners with just a 4% difference.

3.2 DISADVANTAGES OF EXISTING SYSTEM:

- The existing system is not reliable, hence is not recommended for real time applications
- The existing system only focuses on the battery performance of wearable devices
- It is very expensive.

3.3 PROPOSED SYSTEM:

In the last decade, there has been a substantial increase in diseases and an ageing population ratio in the country, necessitating the development of an effective health management system (HMS) for people's convenience. This method is not only cost-effective, but also a viable replacement for conventional healthcare. As a result, the pressure on health services is reduced as the elderly and patients with diseases

withdraw from visiting health institutions (hospitals, nursing homes, and so on). To address these issues and offer adequate care to patients, our project aims to create an application that will continuously track patients suffering from a variety of diseases, including heart rate, temperature, and blood oxidation, both of which will be monitored using accurate sensors. A hardware module with an oxygen and heart rate sensor, a temperature sensor, a GPS module, and a fingerprint scanner makes up the system. The fingerprint scanner is used to authenticate and log users in the database. The information is processed using a real-time microcontroller, and the sensor data is wirelessly transmitted using a Wi-Fi gateway. When a patient's health status becomes urgent, the patient is admitted to the hospital, and the patient's past medical history is shared with hospital administration, as well as the patient's present position, which is determined by the GPS module in the wearable hardware module. The web application is designed to allow users to access patient information, hospital admission bookings, and the patient's current location. As a result, this initiative aids in the timely treatment of patients and the avoidance of problems.

3.4 ADVANTAGES:

- a low-cost and effective method of patient monitoring
- Doctors can easily monitor patients using sensors to identify and predict three parameters: blood oxidation parameter, heart rate, body temperature, and so on.
- The patient's geolocation was correctly identified.
- A web application is being developed to view patient information, hospital admission bookings, and the patient's current location.

3.5 APPLICATIONS:

- Used for test centers
- Used in NGO.
- Used in day care centers
- Used in hospitals.

CHAPTER - 4

MATERIALS AND METHODS

DETAILED DESIGN OF THE PROJECT:

This chapter describes the overall detailed design of the project. It also describes each module.

4.1 BLOCK DIAGRAM:

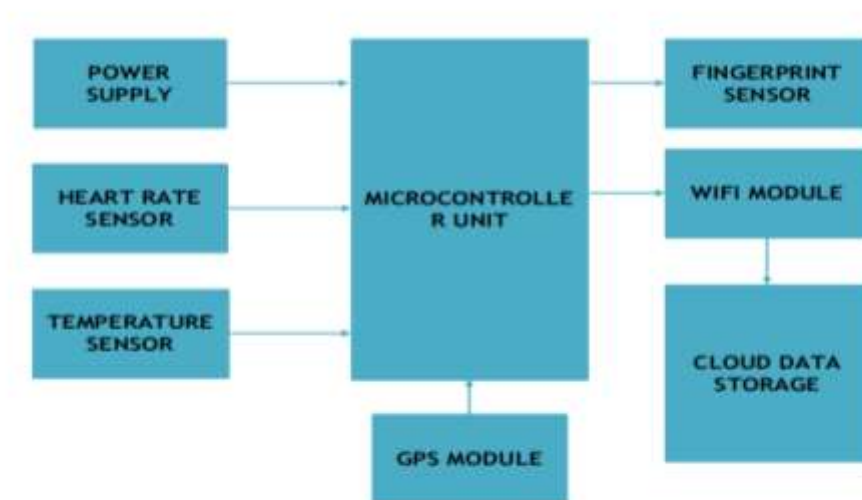


Figure 4.1 Hardware architecture

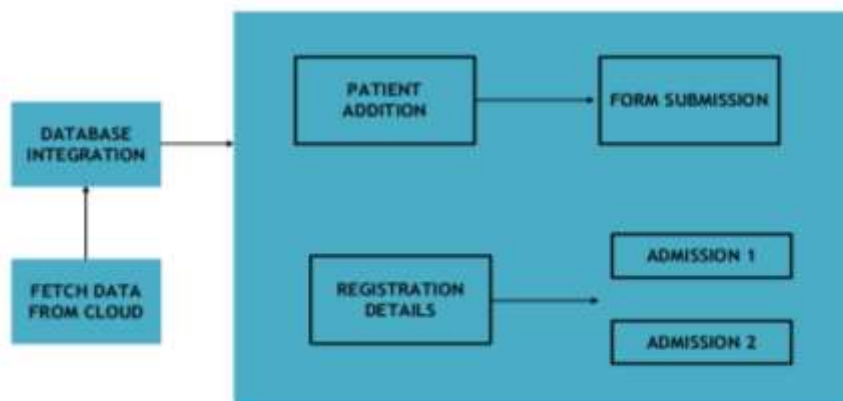


Figure 4.2 System architecture diagram

4.2 WORKING:

The project's goal is to create an application that can continuously monitor patients with various diseases in terms of heart rate, temperature, and blood oxidation, all of which are measured using appropriate sensors. A hardware module with an oxygen and heart rate sensor, a temperature sensor, a GPS module, and a fingerprint scanner makes up the system. The fingerprint scanner is used to authenticate and log users in the database. The information is processed using a real-time microcontroller, and the sensor data is wirelessly transmitted using a Wi-Fi gateway. When a patient's health status becomes urgent, the patient is admitted to the hospital, and the patient's past medical history is shared with hospital administration, as well as the patient's present position, which is determined by the GPS module in the wearable hardware module. The web application is designed to allow users to access patient information, hospital admission bookings, and the patient's current location. As a result, this initiative aids in the timely treatment of patients and the avoidance of problems.

4.3 MODULES DESCRIPTION:

- Power supply module
- Microcontroller module
- Fingerprint sensor
- Temperature sensor
- Heart rate sensor
- MQTT Protocol
- Gateway module
- Web application module

4.3.1 Power Supply Module:

The power supply circuit includes a step-down transformer that converts 230 volts to 12 volts. 4 diodes are used to form a bridge rectifier in this circuit, which delivers pulsating dc voltage and is then fed to a capacitor filter, which eliminates any A.C components present even after rectification.

The filtered DC voltage is fed into the regulator, which produces a constant DC voltage

of 12 volts. 230V AC power is converted to 12V AC (12V RMS value with a peak value of around 17V), but the required power is 5V DC; therefore, 17V AC power must first be converted to DC power before being stepped down to 5V DC. One of the power electronic converters known as a Rectifier can convert AC power to DC. Rectifiers come in a variety of shapes and sizes, including half-wave, full-wave, and bridge rectifiers. Due to the advantages of the bridge rectifier over the half and full wave rectifier, the bridge rectifier is frequently used for converting AC to DC.

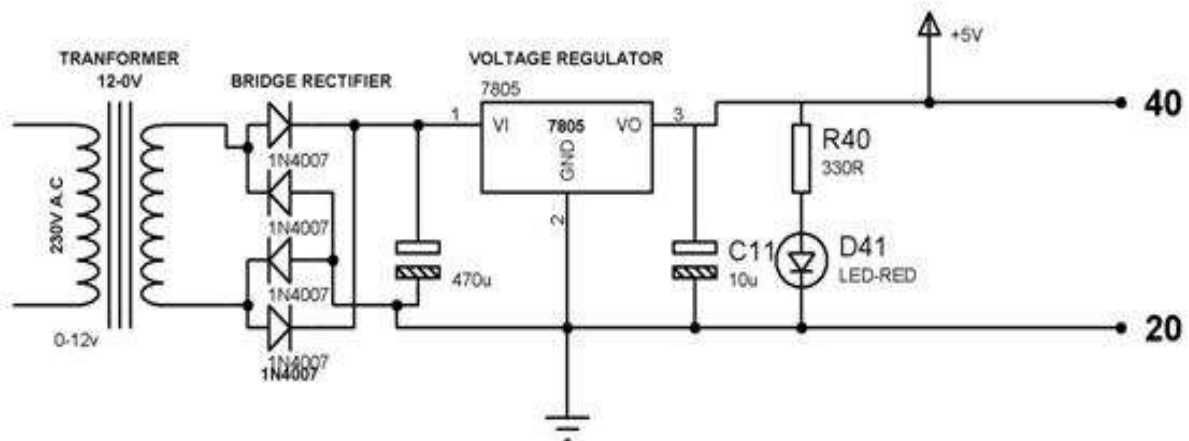


Figure: 4.3 Power supply

4.3.2 Microcontroller Module

The ATmega-328 is a microcontroller based on the Advanced Virtual RISC (AVR) architecture. It can handle data of up to eight (8) bits. The ATmega-328 has 32KB of internal memory. There are a lot of other features on this microcontroller. You should also read Introduction to PIC16F877a (a PIC Microcontroller) before comparing the two Microcontrollers' functions. Electrically Erasable Programmable Read Only Memory (EEPROM) is available on the ATmega 328. (EEPROM). This property indicates whether the microcontroller module is receiving an electric supply. The ATmega-328 is a microcontroller based on the Advanced Virtual RISC (AVR) architecture. It can handle data of up to eight (8) bits. The ATmega-328 has 32KB of internal memory. There are a lot of other features on this microcontroller. You should also read Introduction to PIC16F877a (a PIC Microcontroller) before comparing the two Microcontrollers' functions. Electrically Erasable Programmable Read Only Memory (EEPROM) is available on the ATmega 328. (EEPROM). This property shows if the electric supply supplied to the micro-controller is removed, even then it

can store the data and can provide results after providing it with the electric supply. Furthermore, the ATmega-328 has a 2KB Static Random-Access Memory (SRAM) (SRAM). Other characteristics will be discussed at a later time. The ATmega 328 has a variety of features that make it the most common computer on the market today. Advanced RISC architecture, decent consistency, low power consumption, real timer counter with separate oscillator, 6 PWM pins, programmable Serial USART, programming lock for software protection, throughput up to 20 MIPS, and so on are some of the features. ATmega-328 is mostly used in Arduino.

If the microcontroller is disabled, it will still store data and produce results after being supplied with power. Furthermore, the ATmega-328 has a 2KB Static Random-Access Memory (SRAM) (SRAM). Other traits would be discussed at a later time. The AT mega 328 has a variety of features that make it the most common computer on the market today. Advanced RISC architecture, decent consistency, low power consumption, real timer counter with separate oscillator, 6 PWM pins, programmable Serial USART, programming lock for software protection, throughput up to 20 MIPS, and so on are some of the features. The ATmega-328 is the most commonly used microcontroller in Arduino.

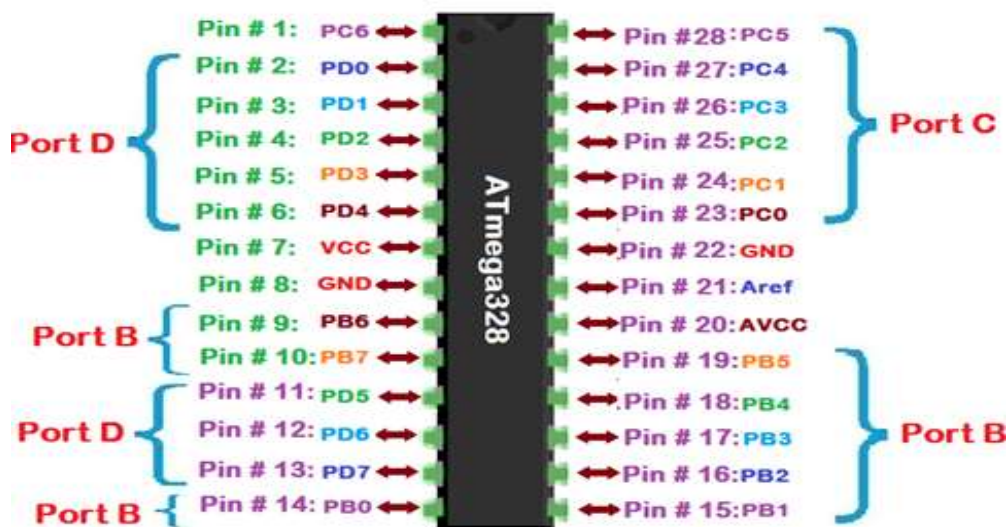


Figure: 4.4 Atmega 328

4.3.3 Fingerprint sensor

The R307 Fingerprint Module is made up of an optical fingerprint sensor, a high-speed DSP processor, a high-performance fingerprint alignment algorithm, high-capacity FLASH chips, and other hardware and software components, with fingerprint entry, image recognition, fingerprint matching, search, and template storage, among other features. The R307 is a TTL UART interfaced finger print sensor board. The user will save the finger print data in the module and use it to identify the individual in 1:1 or 1: N mode. The FP module can communicate directly with a 3v3 microcontroller. Interfacing with a PC necessitates the use of a level converter (such as the MAX232). The R307 fingerprint module has two interfaces: TTL UART and USB2.0. The USB2.0 interface can be attached to a computer; the RS232 interface is a TTL level, with a default baud rate of 57600 that can be modified. And microcontrollers, such as ARM, DSP, and other serial devices with a link, can be directly attached to 3.3V 5V microcontrollers. The machine level conversion, level conversion notice, and embodiments such as a MAX232 circuit must be linked.



Fig 4.5 Fingerprint sensor

4.3.4 Temperature sensor

The LM35 is a kind of temperature sensor that can be used to calculate temperature with an electrical o/p that is proportional to the temperature (in degrees Celsius). In comparison to a thermistor, it can more accurately calculate temperature. Since this sensor produces a higher output voltage than thermocouples, it may not be necessary to amplify the output voltage. The output voltage of the LM35 is proportional to the temperature in Celsius. $0.01\text{V}/^{\circ}\text{C}$ is the scale component. The representation of an LM35 sensor is seen in the diagram.

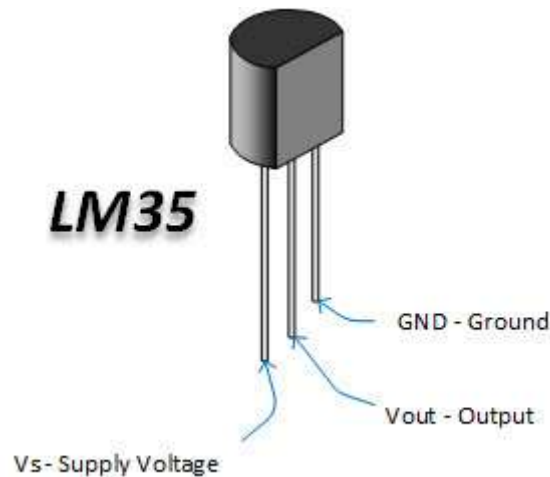


Figure 4.6: Temperature sensor

The LM35 has no outside calibration and has an accuracy of $\pm 0.4^{\circ}\text{C}$ at room temperature and $\pm 0.8^{\circ}\text{C}$ over a temperature range of 0°C to $+100^{\circ}\text{C}$. Another notable feature of this sensor is that it uses only 60 microamps from its power supply and has a limited self-heating capability. TO-46 metal can transistor-like kit, TO-92 plastic transistor-like package, and 8-lead surface mount SO-8 small outline package are all options for the LM35 temperature sensor.

Applications of Lm35 Temperature Sensor

The applications of LM35 temperature sensor include the following

- Measuring temperature of a particular environment and HVAC applications
- Providing thermal shut down for a conductor circuit
- Checking Battery Temperature

4.3.5 Heart Rate Sensor

MAX30100 is a sensor that combines a pulse oximeter and a heart rate monitor. It's an optical sensor that measures the absorbance of pulsing blood through a photodetector after releasing two wavelengths of light from two LEDs – a red and an infrared one. This specific LED colour combination is designed to allow data to be read with the tip of one's finger. The digital output data is stored in a 16-deep FIFO inside the computer, and it is completely configurable via software registers. It communicates with a host microcontroller using an I2C digital interface.

Ambient light cancellation (ALC), 16-bit sigma delta ADC, and patented discrete time filter make up the pulse oximetry subsystem in the MAX30100. It operates at

ultra-low voltage, making it suitable for battery-powered systems. The MAX30100 needs a power supply that is between 1.8 and 3.3 volts. Wearable devices, health assistant devices, medical tracking devices, and other applications will all benefit from it. The MAX30100 uses 1.8V and 3.3V power supplies, and it can be shut down by software with very little standby current, allowing the power supply to stay connected at all times.



Figure: 4.7 MAX30100 sensor

4.3.6 MQTT Protocol

MQTT (MQ Telemetry Transport) is a lightweight, publish-subscribe network protocol that transfers messages between computers. It is an open OASIS and ISO standard (ISO/IEC PRF 20922). MQTT can be used for any network protocol that offers ordered, lossless, bi-directional connections; however, it is most often used for TCP/IP. It's made for remote connections where a "small code footprint" is needed or network capacity is reduced. A message broker and a number of clients are the two types of network entities identified by the MQTT protocol. An MQTT broker is a server

that accepts all messages from clients and forwards them to the correct destination clients. An MQTT client is any system that runs a MQTT library and connects to a MQTT broker over a network (ranging from a microcontroller to a full-fledged server).

The information is arranged in a subject hierarchy. When a publisher has new data to share, it sends a control message to the linked broker along with the data. The information is then distributed to any clients who have subscribed to the subject by the broker. The publisher does not need any information about the number or position of subscribers, and subscribers do not need any information about the publishers.

If a broker receives a subject on which there are no existing subscribers, the broker may discard it unless the publisher specifies otherwise. This encourages potential subscribers to a subject to have the latest up-to-date information without having to wait for a publisher's next update.

When a publishing client connects to the broker for the first time, it will set up a default message that will be transmitted to subscribers if the broker discovers that the publishing client has disconnected suddenly.

Clients only communicate with a broker, but a device can include several broker servers exchanging data based on the topics of their current subscribers. MQTT control messages can be as small as two bytes in size. If necessary, a control message will hold approximately 256 megabytes of data. To bind and detach a client from a broker, to publish data, to accept reception of data, and to supervise the communication between client and server, there are fourteen specified message forms. For data transfer, MQTT uses the TCP protocol. MQTT-SN, a variant, is used over other transports like UDP or Bluetooth.

MQTT sends link credentials in plain text format without any encryption or authentication features. The underlying TCP transport may do this by implementing steps to secure the privacy of transferred data from capture or replication.

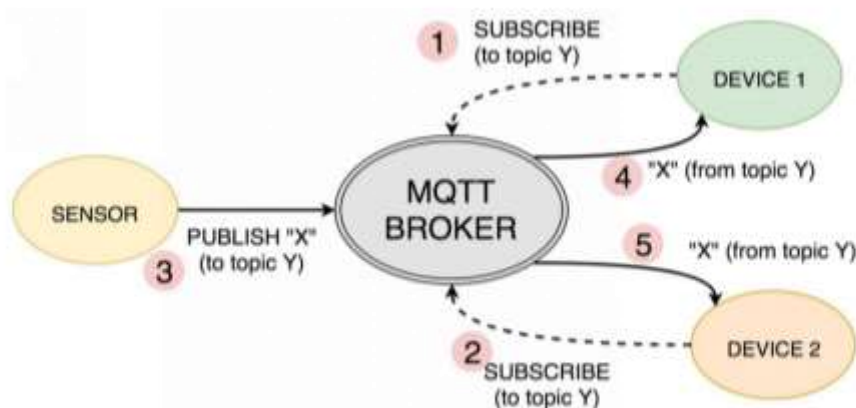


Figure 4.8: MQTT protocol

4.3.7 Gateway Module:

Espresso, a Shanghai-based Chinese maker, produces the ESP8266, a low-cost Wi-Fi chip with complete TCP/IP stack and MCU (microcontroller unit) capabilities. The ESP-01 processor, produced by a third-party designer, AI-Thinker, first brought the chip to the attention of western makers in August 2014. Using Hayes-style instructions, this small module helps microcontrollers to connect to a Wi-Fi network and make basic TCP/IP connections. There was, however, almost no English-language literature on the chip and the commands it recognized at the time. Many hackers were drawn to the module, chip, and applications on it, as well as to translate the Chinese documents, due to the low price and the fact that there were few external components on the module, which meant that it could ultimately be very inexpensive in volume. The ESP8285 is an ESP8266 with 1 MiB of flash built in, allowing single-chip computers to connect to Wi-Fi.

The ESP32 is the successor to these microcontroller chips. Manufacturers use the ESP8266 (currently ESP8266EX) chip to build wirelessly networkable microcontroller modules. ESP8266 is a system-on-a-chip (SoC) with 2.4 GHz Wi-Fi (802.11 b/g/n, supporting WPA/WPA2), general-purpose input/output (16 GPIO), Inter-Integrated Circuit (I2C), analog-to-digital conversion (10-bit ADC), Serial Peripheral Interface (SPI), I2S interfaces with DMA (sharing pins with GPIO), UART (on dedicated pins (PWM)). The Espressif "L106" processor core is built on Tensilica's Diamond Standard 106Micro 32-bit processor controller core and operates at 80 MHz (or overclocked to 160 MHz). It has a 64-kilobyte boot ROM, 64-kilobyte instruction RAM, and 96-kilobyte data RAM. SPI can be used to access external flash memory.



Figure 4.9 ESP8266 module

4.3.8 Web Development

ReactJS is used as a software development platform in this project.

ReactJS is a JavaScript library that allows you to create reusable user interface components. The following is a description from the official React documentation: React is a library for creating composable user interfaces. It supports the creation of interchangeable user interface modules that display data that varies over time.

React is sometimes used as the V in MVC. React abstracts the DOM from you, resulting in a more straightforward programming model and improved performance. React will also use Node to render on the server and React Native to run native applications. One-way reactive data flow is implemented in React, which eliminates boilerplate and makes it simpler to think about than conventional data linking.

JSX

JavaScript XML is abbreviated as JSX. We can write HTML in React using JSX. In React, JSX makes it easy to write and add HTML.

It is faster than standard JavaScript since it optimizes when converting to standard JavaScript. It facilitates the development of models for us. Rather than splitting the markup and logic in different formats, React makes use of components. In subsequent papers, we can go through components in greater depth.

COMPONENTS

Components are self-contained, interchangeable code chunks. They perform the same functions as JavaScript functions, but they operate independently and return HTML with a render function.

Class components and Function components are the two kinds of components; in this tutorial, we'll focus on Class components.

UNIDIRECTIONAL DATA FLOW AND FLUX

One-way data flow is implemented in React, making it simple to reason for the app. Flux is a pattern that aids in the unidirectional flow of data.

ADVANTAGES OF REACT JS

- Using a JavaScript object called virtual DOM. Since JavaScript virtual DOM is faster than the standard DOM, this would increase app speed.
- Can be seen on both the client and server side, as well as in conjunction with other applications.
- Component and data patterns increase readability, allowing for a bigger app to be maintained.

CHAPTER 5

SOFTWARE DESCRIPTION

5.1 VISUAL STUDIO

Microsoft Visual Studio is used as an IDE in this project. Visual Studio Code incorporates the ease of use of a source code editor with advanced developer features such as IntelliSense code completion and debugging.

It is, first and foremost, an editor who stays out of the way. Less time fiddling with the environment and more time executing on your thoughts thanks to the delightfully frictionless edit-build-debug loop.

Visual Studio Code supports macOS, Linux, and Windows - so you can hit the ground running, no matter the platform.

Visual Studio Code's core functionality is a lightning-fast source code editor that's ideal for day-to-day use. Syntax highlighting, bracket-matching, auto-indentation, box-selection, snippets, and more are all available in VS Code, which supports hundreds of languages. You can access the code with ease thanks to intuitive keyboard shortcuts, simple customization, and community-contributed keyboard shortcut mappings.

When it comes to serious coding, you'll also benefit from software that understand more code than just blocks of text. IntelliSense code completion, rich semantic code interpretation and navigation, and code refactoring are all integrated into Visual Studio Code.

When the going gets rough in the computing world, the tough get debugging. We made it possible because debugging is always the one element that developers lack the most in a leaner coding experience. You will walk through source code, inspect variables, display call stacks, and run commands in the console with Visual Studio Code's interactive debugger.

VS Code also combines with create and scripting applications to help you complete basic tasks faster. With Git support in VS Code, you can deal with source control without having to leave the editor, like displaying pending changes diffs.

Customize any feature to your taste and add as many third-party plugins as you like. Although most scenarios work right out of the box, VS Code evolves with you, and we allow you to customise your experience to meet your own needs.

VS Code comes with enhanced built-in support for Node.js creation, including JavaScript and TypeScript, and is fuelled by the same underlying technology as Visual Studio. Web technologies such as JSX/React, HTML, CSS, SCSS, Less, and JSON are all well-represented in VS Code.

Visual Studio Code incorporates the best of internet, native, and language-specific technology in its architecture. VS Code uses Electron to merge web technology like JavaScript and Node.js with native app speed and versatility. VS Code makes use of a newer, quicker version of the same industrial-strength HTML-based editor that powers the “Monaco” cloud editor, Internet Explorer’s F12 Tools, and a variety of other projects. VS Code also uses a tools support framework that allows it to integrate with many of the same technologies that control Visual Studio, such as Roslyn for .NET, TypeScript, the Visual Studio debugging engine, and others.

A public extensibility model in Visual Studio Code allows developers to create and use plugins, as well as configure their edit-build-debug experience.

5.2 ARDUINO

The Arduino software is used in this project. Beginners will find the Arduino app simple to use, but experienced users will find it versatile. It's compatible for Mac, Windows, and Linux. It is used by teachers and students to create low-cost science instruments, to demonstrate chemistry and physics concepts, and to begin learning programming and robotics. Designers and engineers create immersive prototypes, while musicians and performers use it to create sculptures and try out experimental instruments. Makers, for example, use it to build all of the projects on display at the Maker Faire. Arduino is a valuable platform for learning new skills.

- For physical computing, there are a variety of other microcontrollers and microcontroller platforms to choose from. Similar features can be found in Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many other programmes. Both of these programmes condense the complicated aspects of microcontroller programming into an easy-to-use bundle. While Arduino simplifies the process of operating with microcontrollers, it has one benefit over other frameworks for instructors, students, and interested amateurs:

- Low-cost - As opposed to other microcontroller systems, Arduino boards are a bargain. The Arduino module's cheapest variant can be constructed by hand, and even pre-built Arduino modules cost less than \$50.
- Cross-platform - The Arduino Software (IDE) is compatible with Windows, Macintosh OSX, and Linux. The majority of microcontroller systems are only compatible with Windows.
- Easy-to-use programming environment - The Arduino Software (IDE) is simple to use for beginners while still being versatile enough for advanced users. It's based on the Processing programming environment, which is convenient for teachers because students learning to programme in that environment will be familiar with how the Arduino IDE works.
- Open source and extensible software - The Arduino software is available as open-source tools that experienced programmers can extend. People who want to learn more about the language can use C++ libraries, and those who want to learn more about the technical details can switch from Arduino to the AVR C programming language. Similarly, if you want to, you can include AVR-C code directly in your Arduino programme.
- Open source and extensible hardware - The Arduino boards' plans are made available under a Creative Commons licence, allowing experienced circuit designers to create their own version of the module, extending and improving it. The breadboard version of the module can be built by even inexperienced users to learn how it works and save money.

CHAPTER 6

RESULTS AND DISCUSSION

The below image shows the complete hardware setup of the project:



Figure 6.1: Complete hardware setup of the project

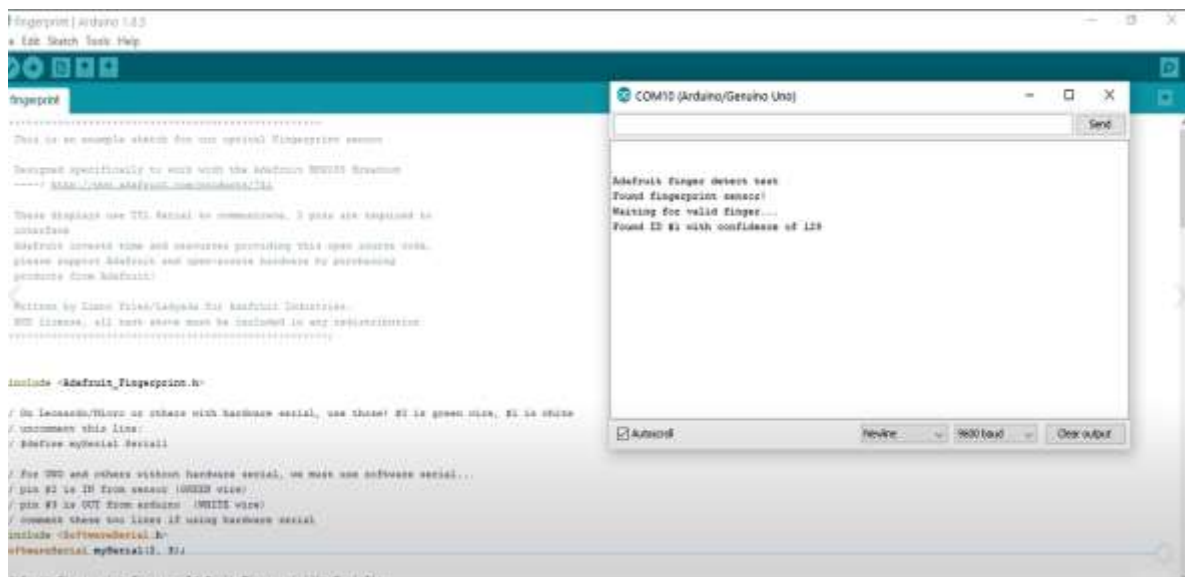


Figure 6.4: Successful fingerprint ID identification

The below image shows the finger being placed on the heart rate sensor to effectively record the heart rate of a patient:



Figure 6.5: Fingerprint placed on the heart rate sensor

The below image shows that heart rate is successfully recorded terming it to be that of a normal heart:

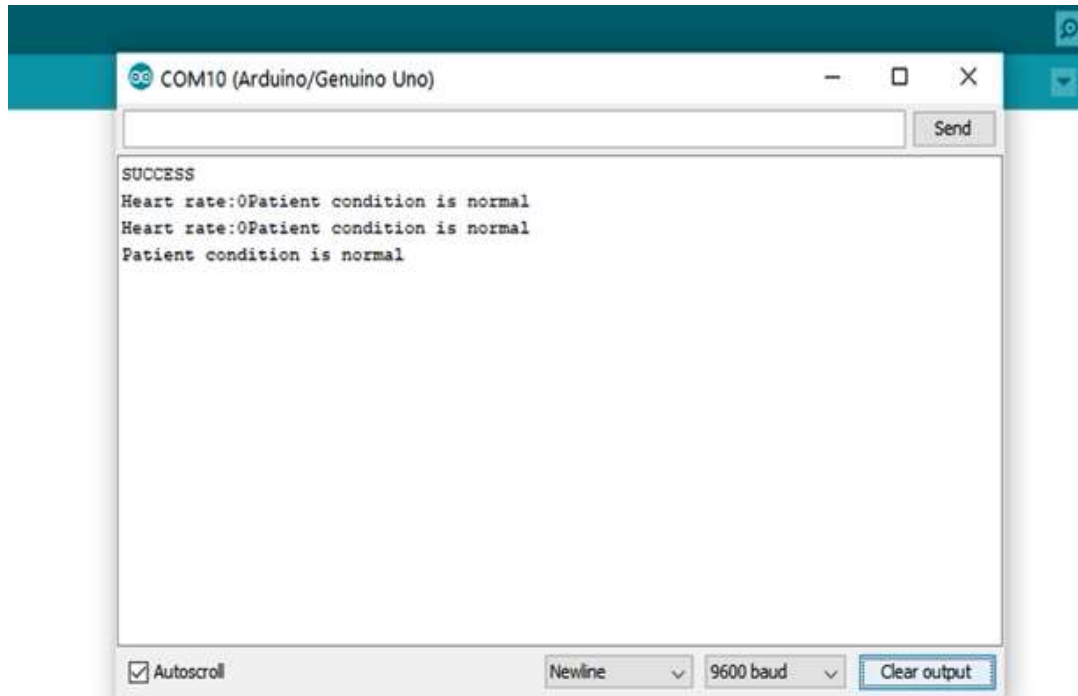


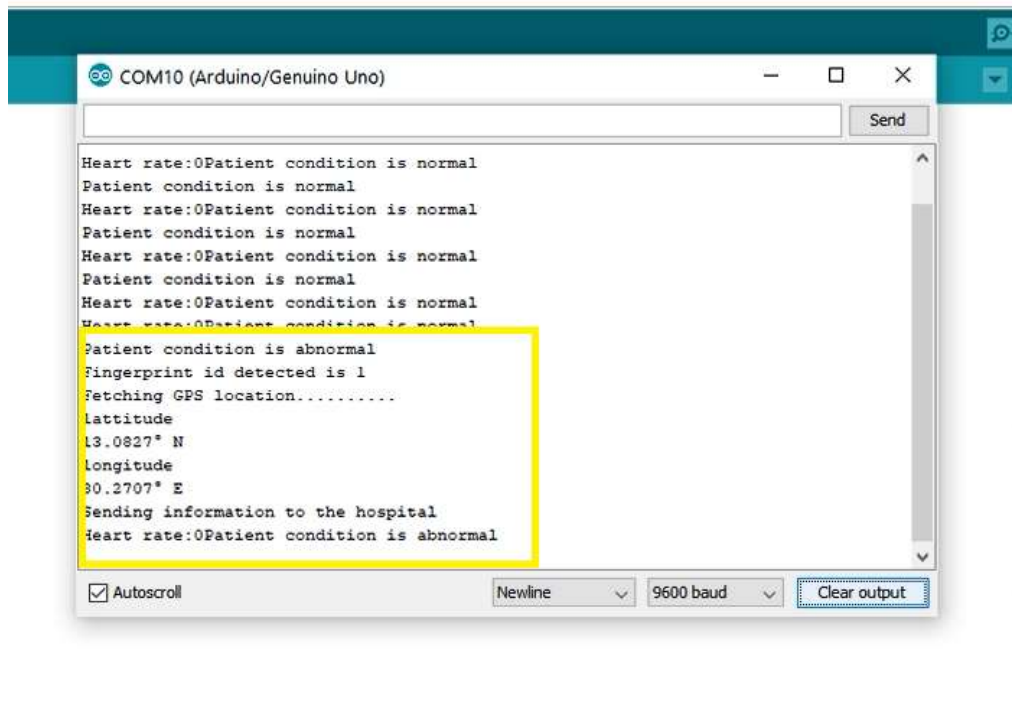
Figure 6.6: Normal heart rate prediction

The below image shows when the patient has removed his/her hand from the heart rate sensor therefore no heart rate should be recorded:



Figure 6.7: Finger removed from the heart rate sensor

The below image shows that the heart rate is not recorded terming it to be that of a abnormal heart:



A web application using react JS is developed for this project. The below image shows the screenshot of the login page of the web application developed:



Figure 6.9: Login page of web application

On successful login the user is routed to the homepage of the web application. The below image shows the patient registration form for the smart ambulance application:

Patient Registration Form

Name (Mr/Ms)	Gender (Male/Female)
Dog#	DOB MM/DD/YYYY
Phone Number (xxx-xxxx-xxxx)	Clinic Name
Veterinarian Name	Address
Referral Hospital	Registration Number (if any)
Weight	City/State
Address	

SUBMIT DATA

Figure 6.10: Patient registration form of web application

Whenever the heart rate, oxidation and temperature of the patient goes abnormal an automatic admission is been done in the hospital which is shown in the below image

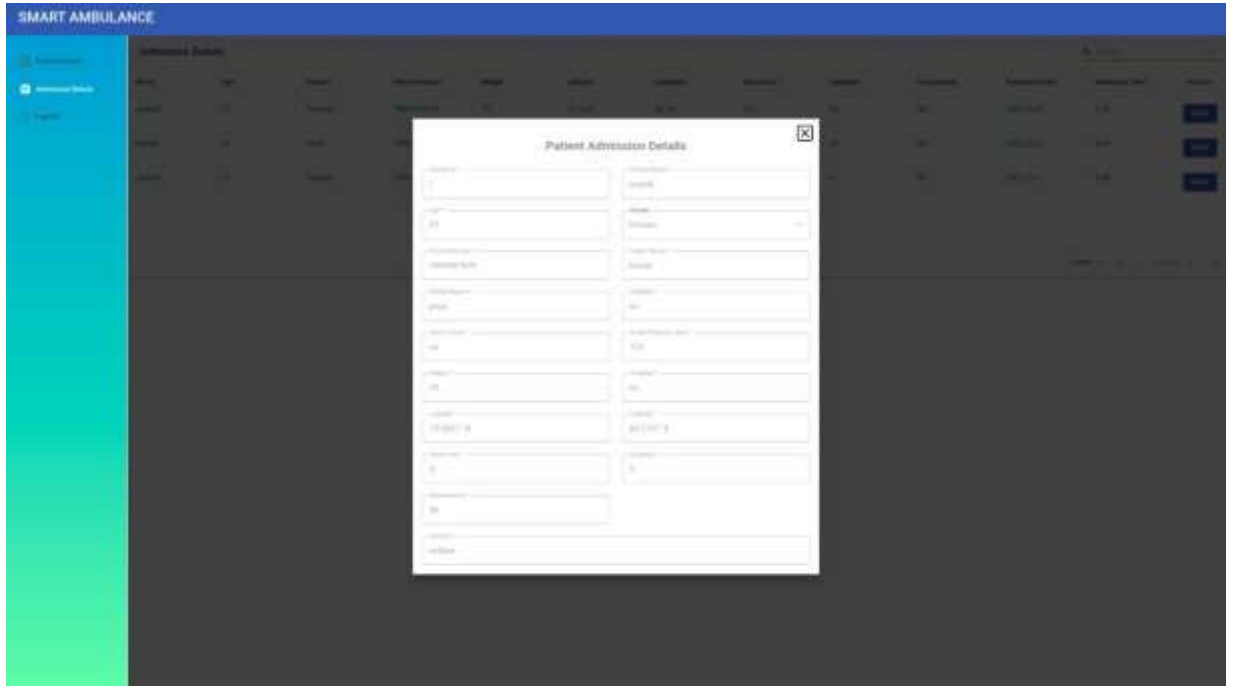
The image is a screenshot of a web application titled "SMART AMBULANCE". On the left, there is a vertical sidebar with a blue-to-green gradient, containing a menu with "Dashboard", "Admission Details", and "Reports". The main area of the application is dark grey and displays a "Patient Admission Details" form as a modal window. The form is white with a title bar and a close button in the top right corner. It contains two columns of input fields for patient information. The first column includes fields for "Patient ID", "Name", "Age", "Gender", "Address", "Phone", "Email", "Blood Group", "Allergy", "Insurance", and "Referral". The second column includes fields for "Admission Date", "Admission Time", "Admission Type", "Admission Status", "Admission Reason", "Admission Location", "Admission Doctor", "Admission Nurse", "Admission Bed", "Admission Room", "Admission Ward", "Admission Unit", "Admission Department", "Admission Hospital", "Admission City", "Admission State", "Admission Country", "Admission Zip", "Admission Latitude", "Admission Longitude", "Admission Altitude", "Admission Temperature", "Admission Heart Rate", "Admission Blood Pressure", "Admission Oxygen Saturation", "Admission Glucose", "Admission Cholesterol", "Admission Hemoglobin", "Admission Hematocrit", "Admission Hemoglobin A1c", "Admission Creatinine", "Admission BUN", "Admission ALT", "Admission AST", "Admission ALP", "Admission GGT", "Admission TBL", "Admission TBA", "Admission TCA", "Admission TCB", "Admission TCD", "Admission TCE", "Admission TCF", "Admission TCG", "Admission TCH", "Admission TCI", "Admission TCU", "Admission TCV", "Admission TCW", "Admission TCX", "Admission TCY", "Admission TCZ", "Admission TC1", "Admission TC2", "Admission TC3", "Admission TC4", "Admission TC5", "Admission TC6", "Admission TC7", "Admission TC8", "Admission TC9", "Admission TC10", "Admission TC11", "Admission TC12", "Admission TC13", "Admission TC14", "Admission TC15", "Admission TC16", "Admission TC17", "Admission TC18", "Admission TC19", "Admission TC20", "Admission TC21", "Admission TC22", "Admission TC23", "Admission TC24", "Admission TC25", "Admission TC26", "Admission TC27", "Admission TC28", "Admission TC29", "Admission TC30", "Admission TC31", "Admission TC32", "Admission TC33", "Admission TC34", "Admission TC35", "Admission TC36", "Admission TC37", "Admission TC38", "Admission TC39", "Admission TC40", "Admission TC41", "Admission TC42", "Admission TC43", "Admission TC44", "Admission TC45", "Admission TC46", "Admission TC47", "Admission TC48", "Admission TC49", "Admission TC50", "Admission TC51", "Admission TC52", "Admission TC53", "Admission TC54", "Admission TC55", "Admission TC56", "Admission TC57", "Admission TC58", "Admission TC59", "Admission TC60", "Admission TC61", "Admission TC62", "Admission TC63", "Admission TC64", "Admission TC65", "Admission TC66", "Admission TC67", "Admission TC68", "Admission TC69", "Admission TC70", "Admission TC71", "Admission TC72", "Admission TC73", "Admission TC74", "Admission TC75", "Admission TC76", "Admission TC77", "Admission TC78", "Admission TC79", "Admission TC80", "Admission TC81", "Admission TC82", "Admission TC83", "Admission TC84", "Admission TC85", "Admission TC86", "Admission TC87", "Admission TC88", "Admission TC89", "Admission TC90", "Admission TC91", "Admission TC92", "Admission TC93", "Admission TC94", "Admission TC95", "Admission TC96", "Admission TC97", "Admission TC98", "Admission TC99", "Admission TC100".

Figure 6.11: Patient admission form of web application

The below image shows the history of the admission details of the already admitted patients:

Name	Age	Gender	Patient Number	Weight	Latitude	Longitude	Heart Rate	Blood Pressure	Temperature	Admission Date	Admission Time	Actions
John Doe	35	Female	1234567890	70	33.44	101.33	98	120	98	2023-10-27	14:00	<button>View</button>
Jane Smith	28	Male	0987654321	85	33.44	101.33	98	120	98	2023-10-27	14:05	<button>View</button>
Robert Johnson	45	Female	1098765432	90	33.5627° N	101.3327° E	95	115	95	2023-10-27	14:10	<button>View</button>

Figure 6.12: Patient admission details tab and Geo location

As a result of the above findings, we have successfully developed a hospital application with an automatic booking system as well as a hardware system that can continuously monitor the patient's condition and take immediate action for registration if the condition becomes abnormal, as well as sharing the live GPS location with the hospital for tracking.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

This proposal proposes a software application creation for a Health Monitoring System to be developed as a consumer electronics device with low computing cost and high accuracy outcomes for automatically monitoring and identifying vital parameters of patients living alone, as well as automatically taking action during abnormality. Furthermore, because of the fast-processing time, this device can be used for all types of patient tracking and is very promising for real-time applications. It is beneficial to patients who are sick.

7.2 FUTURE WORK

This project's web application for a health tracking device can be further improved to be used as a product in real time, and it can be further analysed for greater precision. As a result, applying this in real time would assist all diseased patients in predicting the occurrence of illness at an earlier date.

REFERENCES

- [1] Alexandros Pantelopoulos; Nikolaos G. Bourbakis, "Prognosis—A Wearable Health-Monitoring System for People at Risk: Methodology and Modelling"
- [2] Bruce Wallace, Senior Member, IEEE, Frank Knoefel, Rafifik Goubran, Fellow, IEEE, Philippe Masson, Amanda Baker, Brianna Allard, Victor Guana, and Eleni Stroulia," Detecting Cognitive Ability Changes in Patients with Moderate Cardiomyopathy Using a Modified "Whack-a-Mole" Game, IEEE Transactions on Instrumentation and Measurement, Vol.0018-9456,2017.
- [3] Fasih Haider, Member, IEEE, Sofifia de la Fuente and Saturnino Luz, Member, IEEE," An Assessment of Paralinguistic Acoustic Features for Alzheimer's Cardiomyopathy Recognition in Spontaneous Speech", IEEE Journal of Selected Topics in Signal Processing, Vol. 14, No. 8, August 2016.
- [4] Foad Dabiri; Tammara Massey; Hyduke Noshadi; Hagop Hagopian; C. K. Lin; Robert Tan; Jacob Schmidt; Majid Sarrafzadeh," A Telehealth Architecture for Networked Embedded Systems: A Case Study in In Vivo Health Monitoring"
- [5] Hiroki Tanaka¹ , Hiroyoshi Adachi² , Norimichi Ukita³ , Manabu Ikeda⁴, Hiroaki Kazui⁴ , Takashi Kudo² , And Satoshi Nakamura¹ , (Fellow, Ieee),"Detecting Cardiomyopathy Through Interactive Computer Avatars",IEEE Transactions And Content Mining,Vol.2168-2372,2017.
- [6] Hui Yang, Peter A. Bath," The Use of Data Mining Methods for the Prediction of Cardiomyopathy: Evidence from the English Longitudinal Study of Aging", IEEE Journal of Biomedical and Health Informatics, Vol.2168-2194, 2019.

[7] James Y. Xu; Greg J. Pottie; William J. Kaiser," Enabling Large-Scale Ground-Truth Acquisition and System Evaluation in Wireless Health"

[8] Lei Clifton; David A. Clifton; Marco A. F. Pimentel; Peter J. Watkinson; Lionel Tarassenko," Gaussian Processes for Personalized e-Health Monitoring With Wearable Sensors"

[9] Lijun Zhang, Ming Wang, Nicholas W. Sterling, Eun-Young Lee, Paul J. Eslinger, Daymond Wagner, Guangwei Du, Mechelle M. Lewis, Young Truong, F. DuBois Bowman, Xuemei Huang," Cortical Thinning and Cognitive Impairment in Parkinson's Disease Without Cardiomyopathy",, IEEE/ACM Transactions on Computational Biology and Bioinformatics,Vol.1545-5963,2015.

[10] Mohammad Kachuee, Student Member, IEEE, Mohammad Mahdi Kiani, Student Member, IEEE, Hoda Mohammadzade, Member, IEEE, and Mahdi Shabany, Member, IEEE,"Cuff-Less Blood Pressure Estimation Algorithms for Continuous Health-Care Monitoring",IEEE Transactions on Biomedical Engineering,Vol.0018-9294,2016.

[11] Mitsuru Yoneyama, Hiroshi Mitoma, Nobuo Sanjo, Maya Higuma, Hiroo Terashi, and Takanori Yokota," Ambulatory Gait Behaviour in Patients with Cardiomyopathy: A Comparison with Parkinson's Disease", IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol.1534-4320.2015.

[12] Pholpat Durongbhan, Yifan Zhao, Senior Member, IEEE, Liangyu Chen, Panagiotis Zis, Matteo De, Marco, Zoe C. Unwin, Annalena Venneri, Xiongxiang He, Sheng Li, Yitian Zhao, Daniel J. Blackburn, and Ptolemaios G. Sarrigiannis," A Cardiomyopathy Classification Framework using Frequency and Time-frequency Features based on EEG signals", IEEE Transactions on Neural Systems and Rehabilitation Engineering, Vol.1534-4320,2019.

[13] Sakari Junnila; Harri Kailanto; Juho Merilahti; Antti-Matti Vainio; Antti Vehkaoja; Mari Zakrzewski; Jari Hyttinen," Wireless, Multipurpose In-Home Health Monitoring Platform: Two Case Trials"

[14] Wazir Singh, Student Member, IEEE, Sujay Deb, Member, IEEE, and Pydi Bahubalindrani, Member, IEEE," Low-Noise Energy Efficient Readout Front-End for Wearable Continuous Blood Pressure Monitoring Systems"

[15] Wei-Hsin Wang¹, Yu-Liang Hsu, Member, IEEE, Pau-Choo (Julia) Chung, Fellow, IEEE, and MingChyi Pai," Predictive Models for Evaluating Cognitive Ability in Cardiomyopathy Diagnosis Applications Based on Inertia- and Gait-related Parameters", IEEE Sensors Journal, Vol.1558-1748,2018.

[16] Ting-Ying Li, Yi-Wei Chien, Chi-Chun Chou, Chun-Feng Liao, Member, IEEE, Wen-Ting Cheah, Li-Chen Fu, Fellow, IEEE, Cheryl Chia-Hui Chen, Chun-Chen Chou, and I-A Chen," A Fast and Low-Cost Repetitive Movement Pattern Indicator for Massive Cardiomyopathy Screening", IEEE Transactions on Automation Science Aand Engineering, Vol.1545-5955,2019.

APPENDIX

```
// server.js

// set up
=====

// get all the tools we need
var express = require('express');
var session = require('express-session');
var cookieParser = require('cookie-parser');
var bodyParser = require('body-parser');
var morgan = require('morgan');
var app = express();
var port = process.env.PORT || 4007;
var mysql = require('mysql');
var moment = require('moment');
var dbconfig = require('./config/database');
var connection = mysql.createConnection(dbconfig.connection);
connection.query('USE ' + dbconfig.database);
var pdf = require('pdfkit');
var fs = require('fs');
var http = require('http');
var urlencode = require('urlencode');
var request = require('request');
var mqtt = require('mqtt');
var options = {
    username:'freeza',
    password:'freeza'
};
var client = mqtt.connect('mqtt://broker.hivemq.com',options);
client.on("connect", function () {
});
// function bill_payment_status(paidStatus){
```

```
//          client.publish("freeza/bill_payment",paidStatus);
//          console.log("Publish Command Sent123");
// }
```

```
app.get('/allow',function(req,res){
    let x='#';
    bill_payment_status(x);
    res.redirect('back');
});
```

```
app.get('/disallow',function(req,res){
    let x='*'
    bill_payment_status(x);
    res.redirect('back');
});
```

```
var passport = require('passport');
var flash    = require('connect-flash');
```

```
// configuration
```

```
=====
```

```
// connect to our database
```

```
require('./config/passport')(passport); // pass passport for configuration
```

```
// set up our express application
```

```
app.use(morgan('dev')); // log every request to the console
app.use(cookieParser()); // read cookies (needed for auth)
```

```

app.use(bodyParser.urlencoded({
    extended: true
}));
app.use(bodyParser.json());

app.set('view engine', 'ejs'); // set up ejs for templating

// required for passport
app.use(session({
    secret: 'Mashaallah',
    resave: true,
    saveUninitialized: true
} )); // session secret
app.use(passport.initialize());
app.use(passport.session()); // persistent login sessions
app.use(flash()); // use connect-flash for flash messages stored in session

var today = new Date();
var date = today.getFullYear()+'-'+(today.getMonth()+1)+'-'+today.getDate();
var time = today.getHours() + ":" + today.getMinutes() + ":" + today.getSeconds();

// routes
=====
=====
require('./app/routes.js')(app, passport); // load our routes and pass in our app and
fully configured passport
app.get('/blood_glucose_level',function(req,res){
    connection.query("SELECT * FROM blood_glucose ORDER BY id DESC
LIMIT 1", function(err, rows){
        if(!err){
            res.json(rows);
        }
    });
})

```

```

app.get('/blood_pressure_level',function(req,res){
    connection.query("SELECT * FROM blood_pressure ORDER BY id DESC
LIMIT 1", function(err, rows){
        if(!err){
            res.json(rows);
        }
    });
})

```

```

app.get('/oxidation_level',function(req,res){
    connection.query("SELECT * FROM oxidation ORDER BY id DESC LIMIT 1",
function(err, rows){
        if(!err){
            res.json(rows);
        }
    });
})

```

```

app.get('/graph',function(req,res){
    var end = moment().endOf('month').format('YYYY-MM-DD');
    var start = moment().startOf('month').format('YYYY-MM-DD');
    connection.query("SELECT value as blood_pressure,dt FROM
blood_pressure where sensors_data_date='"+date+"' order by dt asc", function(err,
rows){
        if(!err){
            res.json(rows);
        }
    });
})

```

```

app.get('/graph2',function(req,res){
    var end = moment().endOf('month').format('YYYY-MM-DD');
    var start = moment().startOf('month').format('YYYY-MM-DD');

```

```

        connection.query("SELECT value as blood_glucose,dt FROM
blood_glucose where sensors_data_date='"+date+"' order by dt asc", function(err,
rows){
            if(!err){
                res.json(rows);
            }
        });
    })

```

```

app.get('/graph3',function(req,res){
    var end = moment().endOf('month').format('YYYY-MM-DD');
    var start = moment().startOf('month').format('YYYY-MM-DD');
    connection.query("SELECT value as oxidation,dt FROM oxidation
where sensors_data_date='"+date+"' order by dt asc", function(err, rows){
        if(!err){
            res.json(rows);
        }
    });
})

```

// launch

```

=====
=====

```

```

app.listen(port);
console.log('The magic happens on port ' + port);

```

```

var mqtt = require('mqtt');
var mysql = require('mysql');
var urlencode = require('urlencode');
var http = require('http');
var dbconfig = require('./config/database');
var options = {
    username:'freeza',
    password:'freeza'

```

```

};

var connection = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  //password: 'n5sYGPfvr7hunpxR',
  database: 'Cardiomyopathy_monitor'
});
connection.connect();

function alertSms(x){

  alertDetails = x+' is Abnormal Kindly Check..Take Immediate Action..';

var msg = urlencode(alertDetails);
var toNumber = '8870986858';
var username = 'yadavaprasath@gmail.com';
var hash =
'09d4e4bf05e0cb189844bd927fdbe56296338c77b04cf2f4f3f1c54077118e44'; // The
hash key could be found under Help->All Documentation->Your hash key.
Alternatively you can use your Textlocal password in plain text.
var sender = 'txtlcl';
var data = 'username=' + username + '&hash=' + hash + '&sender=' + sender +
'&numbers=' + toNumber + '&message=' + msg;
var options = {
  host: 'api.textlocal.in', path: '/send?' + data
};

callback = function (response) {
  var str = "";//another chunk of data has been recieved, so append it to `str`
  response.on('data', function (chunk) {
    str += chunk;
  });//the whole response has been recieved, so we just print it out here
  response.on('end', function () {

```

```

    console.log(str);
  });
} // console.log('hello js'))
http.request(options, callback).end(); // url encode instalation need to use $ npm
install urlencode

}

function alertcheck(){

    connection.query("SELECT * FROM blood_pressure ORDER BY id DESC
LIMIT 1", function(err, rows){
        blood_pressure = rows[0].level;

        connection.query("SELECT * FROM blood_glucose ORDER BY id DESC LIMIT
1", function(err, rows){
            blood_glucose = rows[0].level;

            connection.query("SELECT * FROM oxidation ORDER BY id DESC LIMIT 1",
function(err, rows){
                oxidation = rows[0].level;

                console.log("Blood pressure level is " + blood_pressure);
                console.log("Blood glucose level is " + blood_glucose);
                console.log("Oxidation level is" + oxidation);

                if (blood_pressure=="ABNORMAL" && blood_glucose=="ABNORMAL" &&
oxidation=="ABNORMAL"){

```

```

        alertSms("Blood Pressure, Blood Glucose and Oxidation level");

    }
    else {
        console.log("No cardiomyopathy found");

    }

});
});
});

}

```

```

var client = mqtt.connect('mqtt://broker.hivemq.com',options);
//connection mysql
var connection = mysql.createConnection(dbconfig.connection);
connection.query('USE ' + dbconfig.database);

//Connecting Mqtt
client.on('connect', function () {
    client.subscribe('cardiomyopathy/oxidation');
    console.log(`MQTT Started to Listen...`);
});

client.on('message', function (topic, message) {
    // message is Buffer

```

```

var recievedMessage = message.toString();
console.log(recievedMessage);

let checkflag;
connection.query("SELECT `value` FROM oxidation ORDER BY id DESC
LIMIT 1", function(err,rows){
    if(err){
        console.log("Error getting previous oxidation value ",err);
    }else{
        if(rows[0].value == recievedMessage || rows[0].value !=
recievedMessage){
            checkflag = true;
        }else{
            checkflag = false;
        }

        if(checkflag == true){
            if(recievedMessage >= 70 && recievedMessage <= 120){
                // alertSms("oxidation");

                var today = new Date();
                var date = today.getFullYear()+'-'+(today.getMonth()+1)+'-'
'+today.getDate();
                var time = today.getHours() + ":" + today.getMinutes() + ":" +
today.getSeconds();

                connection.query('INSERT INTO `oxidation`(
`level`,`value`,`sensors_data_date`,`sensors_data_time`) VALUES
("NORMAL","'+recievedMessage+'","'+date+'","'+time+'")', function(err){
                    if(err){
                        console.log("Error inserting oxidation Level ",err);
                    }else{
                        console.log("NORMAL oxidation data inserted successfully");

```

```

        console.log(date);
        console.log(time);
    }
})
}else {

    var today = new Date();
    var date = today.getFullYear()+'-'+(today.getMonth()+1)+'-'
+today.getDate();
    var time = today.getHours() + ":" + today.getMinutes() + ":" +
today.getSeconds();

    connection.query('INSERT INTO `oxidation`(
`level`,`value`,`sensors_data_date`,`sensors_data_time`) VALUES
("ABNORMAL",""+recievedMessage+"",""+date+"",""+time+"")', function(err){
        if(err){
            console.log("Error inserting oxidation Level ",err);
        }else{
            console.log("ABNORMAL oxidation data inserted successfully");
        }
    })

    }

    }
}

});

var client1 = mqtt.connect('mqtt://broker.hivemq.com');
client1.on('connect', function () {
    client1.subscribe('cardiomyopathy/glucose2');

```

```
});
```

```
client1.on('message', function (topic, message) {  
  // message is Buffer  
  var recievedMessage = message.toString();  
  console.log("Test...",recievedMessage);
```

```
  let checkflag;  
  connection.query("SELECT `value` FROM blood_glucose ORDER BY id DESC  
LIMIT 1", function(err,rows){  
    if(err){  
      console.log("Error getting previous blood_glucose value ",err);  
    }else{  
      if(rows[0].value == recievedMessage || rows[0].value != recievedMessage  
) {  
        checkflag = true;  
      }else{  
        checkflag = false;  
      }  
  
      if(checkflag == true){  
        if(recievedMessage == 0 ){  
          //alertSms("glocose");  
  
          var today = new Date();  
          var date = today.getFullYear()+'-'+(today.getMonth()+1)+'-'  
+today.getDate();  
          var time = today.getHours() + ":" + today.getMinutes() + ":" +  
today.getSeconds();  
  
          connection.query('INSERT INTO `blood_glucose`(  
`level`,`value`,`sensors_data_date`,`sensors_data_time`) VALUES  
("NORMAL","'+recievedMessage+'","'+date+'","'+time+'")', function(err){
```

```

        if(err){
            console.log("Error inserting blood glucose Level ",err);
        }else{
            console.log("Normal blood glucose data inserted successfully");
        }
        alertcheck();
    })
    }else {

        var today = new Date();
        var date = today.getFullYear()+'-'+(today.getMonth()+1)+'-'
'+today.getDate();
        var time = today.getHours() + ":" + today.getMinutes() + ":" +
today.getSeconds();

        connection.query('INSERT INTO `blood_glucose` (
`level`,`value`,`sensors_data_date`,`sensors_data_time`) VALUES
("ABNORMAL","'+recievedMessage+'","'+date+'","'+time+'")', function(err){
            if(err){
                console.log("Error inserting blood glucose Level ",err);
            }else{
                console.log("Abnormal blood glucose data inserted successfully");
            }
            alertcheck();
        })

    }

}

})

```

```

});

var client2 = mqtt.connect('mqtt://broker.hivemq.com');
client2.on('connect', function () {
    client2.subscribe('cardiomyopathy/pressure2');
});

client2.on('message', function (topic, message) {
    // message is Buffer
    var recievedMessage = message.toString();
    console.log(recievedMessage);

    let checkflag;
    connection.query("SELECT `value` FROM blood_pressure ORDER BY id
DESC LIMIT 1", function(err,rows){
        if(err){
            console.log("Error getting previous blood_pressure value ",err);
        }else{
            if(rows[0].value == recievedMessage || rows[0].value !=
recievedMessage){
                checkflag = true;
            }else{
                checkflag = false;
            }

            if(checkflag == true){
                if(recievedMessage ==0){
                    // alertSms("blood pressure");
                    var today = new Date();
                    var date = today.getFullYear()+'-'+(today.getMonth()+1)+'-'
'+today.getDate();
                    var time = today.getHours() + ":" + today.getMinutes() + ":" +
today.getSeconds();

```

```

        connection.query('INSERT INTO `blood_pressure`(`level`,`value`,`sensors_data_date`,`sensors_data_time`) VALUES ("NORMAL","'+recievedMessage+'","'+date+'","'+time+'")', function(err){
            if(err){
                console.log("Error inserting blood pressure Level ",err);
            }else{
                console.log("Normal blood pressure data inserted successfully");
            }
        })
    }else {

```

```

        //alertSms("blood pressure");

```

```

        var today = new Date();
        var date = today.getFullYear()+'-'+(today.getMonth()+1)+'-'+today.getDate();
        var time = today.getHours() + ":" + today.getMinutes() + ":" + today.getSeconds();

```

```

        connection.query('INSERT INTO `blood_pressure`(`level`,`value`,`sensors_data_date`,`sensors_data_time`) VALUES ("ABNORMAL","'+recievedMessage+'","'+date+'","'+time+'")', function(err){
            if(err){
                console.log("Error inserting blood pressure Level ",err);
            }else{
                console.log("Abnormal blood pressure data inserted successfully");
            }
        })
    }

}

}

```

})

});