HEALTHCARE MONITORING AND MANAGEMENT SYSTEM

by Souvik Karmakar

A minor project report on :

HEALTHCARE MONITORING AND MANAGEMENT SYSTEM

submitted in partial fulfilment of the requirements for the degree of
B. Tech
In
Electronics and Electrical Engineering
By

Souvik Karmakar	**1807228**
Indrashis Mitra	**1807274**
Kinjal Sarkar	**1807277**
Pratyay Basu	**1807291**

under the guidance of
Prof. K.B. Ray

School of Electronics Engineering
KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY
(Deemed to be University)
BHUBANESWAR

April 2021

CERTIFICATE

This is to certify that the project report entitled **"HEALTHCARE MONITORING AND MANAGEMENT SYSTEM"** submitted by

Souvik Karmakar	**1807228**
Indrashis Mitra	**1807274**
Kinjal Sarkar	**1807277**
Pratyay Basu	**1807291**

in partial fulfilment of the requirements for the award of the **Degree of Bachelor of Technology** in **Electronics and Electrical Engineering** is a bonafide record of the work carried out under my(our) guidance and supervision at School of Electronics Engineering, KIIT (Deemed to be University).

Signature of Supervisor 1
Prof.K.B.Ray
School of Electronics Engineering
KIIT (Deemed to be University)

The Project was evaluated by us on _____________

EXAMINER 1　　　　EXAMINER 2

EXAMINER 3　　　　EXAMINER 4

ACKNOWLEDGEMENT

We feel immense pleasure and feel privileged in expressing our deepest and most sincere gratitude to our supervisor **Prof.K.B.Ray**, for her excellent guidance throughout our project work. Her kindness, dedication, hard work and attention to detail have been a great inspiration to us. Our heartfelt thanks to you mam for the unlimited support and patience shown to us. We would particularly like to thank her for all her help in patiently and carefully correcting all our manuscripts.

We are also very thankful to **Prof.Suman Roy, Prof. Pravat Biswal**
Prof. M. Ramana, FIC project (EEE), Associate Dean Prof. **Dr. Amlan Datta** and **Dr. Suprava Patnaik**, Dean (School Of Electronics) for their support and suggestions during our course of the project work in the final year of our undergraduate course.

STUDENT SIGNATURE :-

Roll Number	Name	Signature
1807228	**Souvik Karmakar**	Souvik Karmakar
1807274	**Indrashis Mitra**	Indrashis Mitra
1807277	**Kinjal Sarkar**	Kinjal Sarkar
1807291	**Pratyay Basu**	Pratyay Basu

Date:-

ABSTRACT

In the present world, IoT is changing the infrastructure of technologies. Internet of Things (IoT) based smart health monitoring system is a patient monitoring system in which a patient can be monitored 24 hours. Remote Patient Monitoring arrangement empowers observation of patients outside of customary clinical settings (e.g. at home), which expands access to human services offices at bring down expenses. Healthcare is given extreme importance now a-days by each country with the advent of the novel corona virus. With the increase in use of wearable sensors and the smart phones, remote health care monitoring has also evolved. IoT monitoring of health helps in preventing the spread of disease as well as to get a proper diagnosis of the state of health, even if the doctor is at a far distance. By facilitating effortless interaction among various modules, IoT has enabled us to implement various complex systems such as smart home appliances, smart traffic control systems, smart office systems, smart environment, smart vehicles and smart temperature control systems and so on in very little space. Health monitoring systems are one of the most notable applications of IoT. Many types of designs and patterns have already been implemented to monitor a patient's health condition through IoT.

Several life-threatening diseases can be easily monitored by IoT based systems. Cardiovascular Disease (CVD) is a common disease which is the cause behind most of the deaths in the world. The number of heart bits per minute is denoted as the heart rate of the patient. It is also referred to as the pulse rate of the body. The rate changes with illness, due to damage to body, heart, and exercise. Hence heart rate is essential in determining one's health condition.

PROPOSED SYSTEM: The core objective of this project is the design and implementation of a smart patient health tracking system. The sensors are embedded on the patient body to sense the temperature and heartbeat of the patient. These sensors are connected to a control unit, which calculates the values of all the sensors. These calculated values are then transmitted through a IoT cloud to the base station. From the base station the values are then accessed by the doctor at any other location. Thus based on the temperature and heart beat values, the doctor can decide the state of the patient and appropriate measures can be taken.

TABLE OF CONTENTS

Abstract

Table of Contents

List of Figures

List of Tables

List of symbols/ abbreviations

CHAPTER I: INTRODUCTION 1

1.1 Background 1

1.2 Organization of report 1

CHAPTER 2: EMBEDDED SYSTEMS - Metrics Monitoring

2.1 Objective (I) and Components required 2

(a) Detailed description of the components 3

(b) Module features 3

2.2 MLX90614 detailed pin configuration 3

2.3 MLX90614 -Features and benefits 4

2.4 Advantages of MLX90614 4

2.5 Applications of MLX90614 5

2.6 Arduino UNO interfacing of MLX90614 6-7

2.7 Board schematic representation 8

2.8 MAX30102 Pulse Rate Sensor - Product description 9

2.9 Detailed pin description of MAX30102 10

2.10 Advantages of MAX30102 10

2.11 Principle of working and specifications 11

2.12	Arduino UNO Interfacing MAX30102	12
2.13	Flowchart of the procedure.	13

CHAPTER 3: **EMBEDDED SYSTEMS - FALL DETECTION SYSTEM**

3.1	Objective (II) and components required	14
3.2	Sensor Description (MPU6050)	15
3.3	Pin description of MPU6050	16
3.4	Using procedure of MPU6050	17-18
3.5	MPU6050 features	18
3.6	Interfacing of MPU6050 with Arduino UNO	19
3.7	Model activity	20-21
3.8	Data Acquisition	22-23
3.9	Flow diagram	23
3.10	IFTTT integration for the fall detection system	24-27
3.11	Connection diagram	28

CHAPTER 4: **IOT - WEBSITE MONITORING**

4.1	Introduction to Thinger.io	29-30
4.2	Main features of Thinger.io	30-31
4.3	Working procedure	32-36

CHAPTER 5: **IOT- BLYNK**

5.1	Introduction to the Blynk Community	37
5.2	Blynk features	38
5.3	Introduction to Blynk account and interface	39-42
5.4	Blynk application in metrics monitoring	42

5.5 Blynk - My APPS 43-44

CHAPTER 6:CONCLUSION

6.1: Summary 45

6.2: Future scope 46-48

6.3 Cost Analysis 49

CHAPTER 7 : PLANNING & REFERENCES 50-52

LIST OF FIGURES

Fig no.	Description	Page No.
2.1	MLX90614 temperature sensor module	2
2.2	MLX90614 Pin-out Configuration	3
2.3	Arduino UNO and Nodemcu interfacing with MLX90614	6
2.4	Connection diagram of MLX90614 with nodemcu board	7
2.5	Board schematic representation	8
2.6	MAX 30102 pulse rate sensor	9
2.7	MAX 30102 Pin out configuration	10
2.8	Connection diagram of MAX30102 with Nodemcu	12
2.9	Flow diagram (For monitoring parameters)	13
3.1	MPU6050 - Accelerometer and Gyroscope Module	14
3.2	MPU6050 PINOUT	16
3.3	Board schematic	17

3.4	Circuit Connection of MPU6050	18
3.5	Connection of MPU6050 sensor with Nodemcu board	19
3.6	Axis representation of gyroscope (MPU6050)	20
3.6(a)	The placement of the sensor board	22
3.6(b)	The geodetic coordinate OXYZ	22
3.7(a & b)	Front & back view of sensor board with Bluetooth, tri-axial accelerometer and gyroscope.	23
3.8	Flow diagram	24
3.9	Illustration of IFTTT steps	27
3.10	Connection diagram	28
4.1	Connectivity of thinger.io platform	31
4.2	Illustration of widget selection	33
4.3	Dashboard layout	34
4.4(a &b)	An example of how a dashboard monitor looks like	36
5.1	Organization of Blynk cloud	38
5.2	Illustration of account creation	39
5.3	Creation of project	40
5.4	Selection of hardware	41
5.5	Widget selection	41
5.6	Running the project	42
5.7	Blynk MyApp layout	44
6.1	Gantt Chart	50

LIST OF TABLES

Table ID	Table Title	Page
3.1	MLX 90614 Pin out configuration	7
3.2	Pin connection table of MLX90614	10
3.3	Pin connection table of MAX 30102	12
3.4	Pin Connection chart of MAX 30102 with Nodemcu board	15-16
6.3	Cost Analysis	48
7.1	Showing details about project planning and management	49

LIST OF SYMBOLS / ABBREVIATIONS

Symbol / Abbreviations	Description
MLX	Melexis
MEMS	Micro-electromechanical systems
GPRS	General Packet Radio Service
Wifi	**Wireless Fidelity**
MAX	Maxim integrated
ESP	Espressif Systems
IC	Integrated Circuit
I2C	Inter-Integrated Circuit

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND

At present, with the revolution of information and technology, smartphone-based health monitoring systems are becoming more popular. These systems can be used to collect real-time health information and give feedback to patients and medical specialists. Allowing every single person to examine their health, and advising them to find immediate treatment in case of emergencies, can result in saving that person's life. The use of these monitoring systems can decrease medical fees for the nation in the long run. Nowadays, due to widespread mobile internet access, the combination of mobile internet with a health service system using android open-source design has become very easy. Using a smart device, doctors and patients can continuously observe the heart rate and can get important data and take proper steps to prevent severe damages . Heart rate and body temperature are some of the most important traits of the human body which are major contributors to determining a patient's health condition.

The normal pulse rate of a healthy adult is 60 to 100 beats per minute. The average human pulse rate is 70 beats per minute for males and 75 beats per minute are for females. Females aged 12 and older have faster heart rates than males.

1.2 ORGANIZATION OF THE REPORT

This report has been divided into 7 chapters: -

Chapter 1 – Introduction

Chapter 2 – Embedded systems – Metrics monitoring

Chapter 3 – Embedded systems – Fall detection system

Chapter 4 – IOT – Website monitoring

Chapter 5 – IOT – Blynk

Chapter 6 – Conclusion & Future Scope

Chapter 7 – Planning & References

CHAPTER 2

EMBEDDED SYSTEMS: METRICS MONITORING

OBJECTIVE: Monitor patient parameters remotely to increase efficacy of healthcare management systems

IDEATION:

We need to monitor the patient parameters from remote distances using various sensors. The data given out by the sensors are then sent over to cloud for further access via a Wi-Fi module (inbuilt or externally connected). This is being done to reduce the critical time of testing patient parameters before any major operation.

Here we are measuring the temperature and pulse parameter of a patient remotely.

2.1 Components required :-

1. MLX 90614 Temperature Sensor.
2. MAX 30102 Pulse Rate Sensor.
3. Jumper wires.
4. Nodemcu ESP8266 board

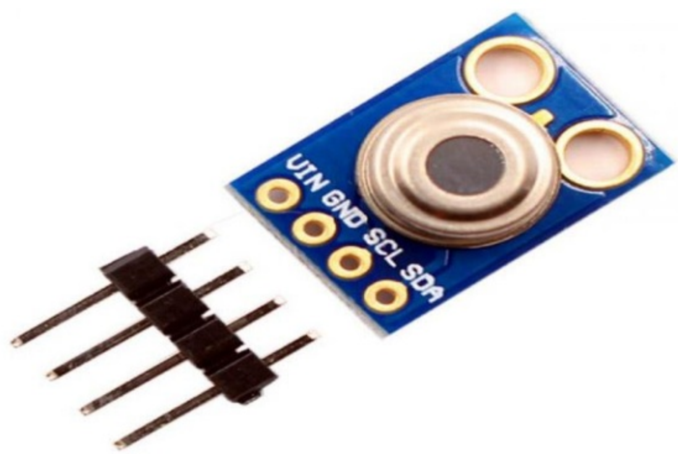


Fig 2.1 MLX90614 temperature sensor module

(a) Product Description: - (MLX90614)

The MLX90614 is a **Contact-less Infrared (IR) Digital Temperature Sensor** that can be used to measure the temperature of a particular object ranging from -70° C to 382.2°C. The sensor uses IR rays to measure the temperature of the object without any physical contact and communicates to the micro-controller using the I2C protocol.

The MLX90614 is a **Contact less Infrared (IR) Digital Temperature Sensor** that can be used to measure the temperature of a particular object ranging from -70° C to 382.2°C. The sensor uses IR rays to measure the temperature of the object without any physical contact and communicates to the micro-controller using the I2C protocol.

(b) Module Features:-

- Works from 3.3V to 5V input, Module has power regulator IC built in.
- Standard I2C interface with built 2x pull up resistors
- Operating Voltage: 3V- 5V
- Operating Current: 2mA
- Communication protocol: I2C.
- Sensor working temperature: -40 to +125 ° C
- Sensing temperature range: -70 to +380 °C
- When measuring the temperature, please maintain a measuring distance of 1 cm
- PCB dimensions 11mm x 17 mm

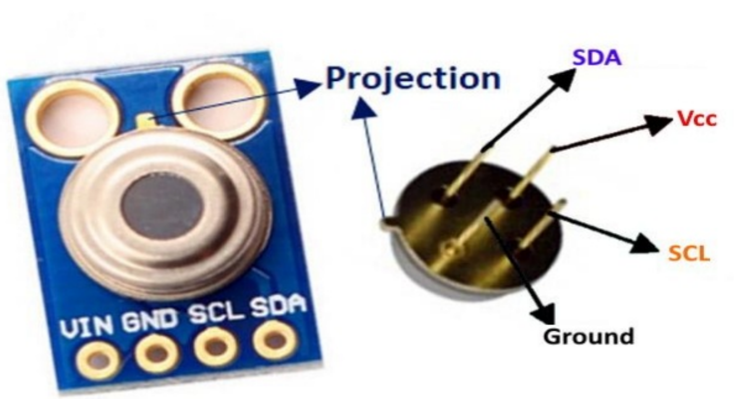


Fig 2.2 MLX90614 Pin-out Configuration

Pin No.	Pin Name	Description
1	**V_{dd} (Power supply)**	V_{dd} can be used to power the sensor, typically using 5V
2	**Ground**	The metal can also act as ground
3	**SDA – Serial Data**	Serial data pin used for I2C Communication
4	**SCL – Serial Clock**	Serial Clock Pin used for I2C Communication

Table 3.1

2.3 MLX90614 Features and Benefits

- Small size, low cost
- Easy to integrate
- Factory calibrated in wide temperature range: -40 to 125 °C for sensor temperature and -70 to 380 °C for object temperature.
- High accuracy of 0.5°C over wide temperature range (0 - +50 °C for both T_a and T_o)
- Medical accuracy of 0.1°C in a limited temperature range available on request
- Measurement resolution of 0.01°C
- SMBus compatible digital interface for fast temperature readings and building sensor networks
- Customizable PWM output for continuous reading
- Simple adaptation for 8 to 16V applications
- Power saving mode
- Automotive grade

2.4 Advantages of MLX90614 over other Temp sensors:-

- Our project is related to the medical field so we chose the temp sensor which gives good accuracy. MLX 90614 's accuracy is near about 0.02° C so it is far better than other temp sensors.
- It is a contact-less infrared temp sensor and in this pandemic situation contact-less is the best option. It can sense the temp of an object from a distance of 2cm to 5cm approx.
- Its object temp range is near about -70 to 380 °C, so it can sense the wide range of temp.
- MLX 90614 consumes low voltage and low current. Its operating voltage is near about 3 V to 5 V and the required supply current is near about 2 mA.

2.5 Applications:-

- High precision non-contact temperature measurements
- Thermal Comfort sensor for Mobile Air Conditioning control system
- Temperature sensing element for residential, commercial and industrial building air conditioning Wind shield defogging
- Automotive blind angle detection
- Industrial temperature control of moving parts
- Temperature control in printers and copiers
- Home appliances with temperature control
- Healthcare
- Livestock monitoring
- Movement detection
- Multiple zone temperature control – up to 127sensors can be read via common 2 wires
- Thermal relay / alert
- Body temperature measurement

2.6

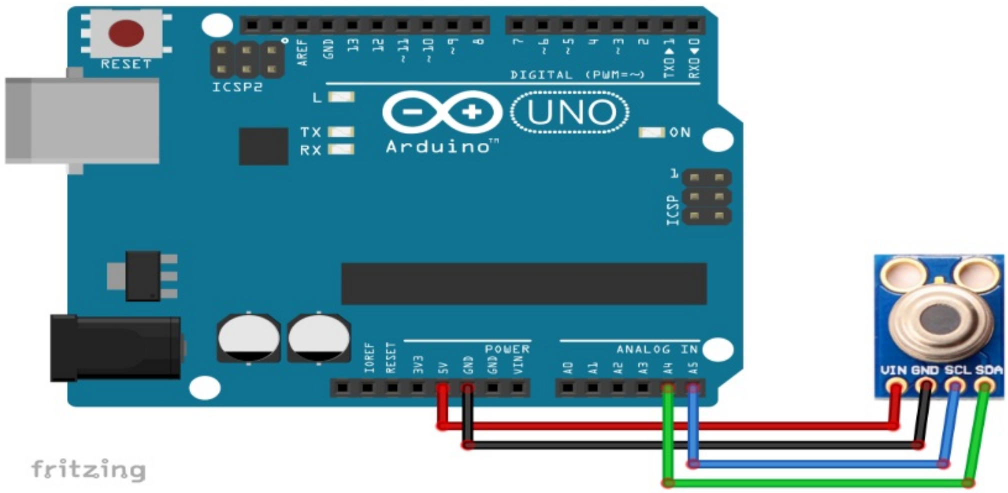
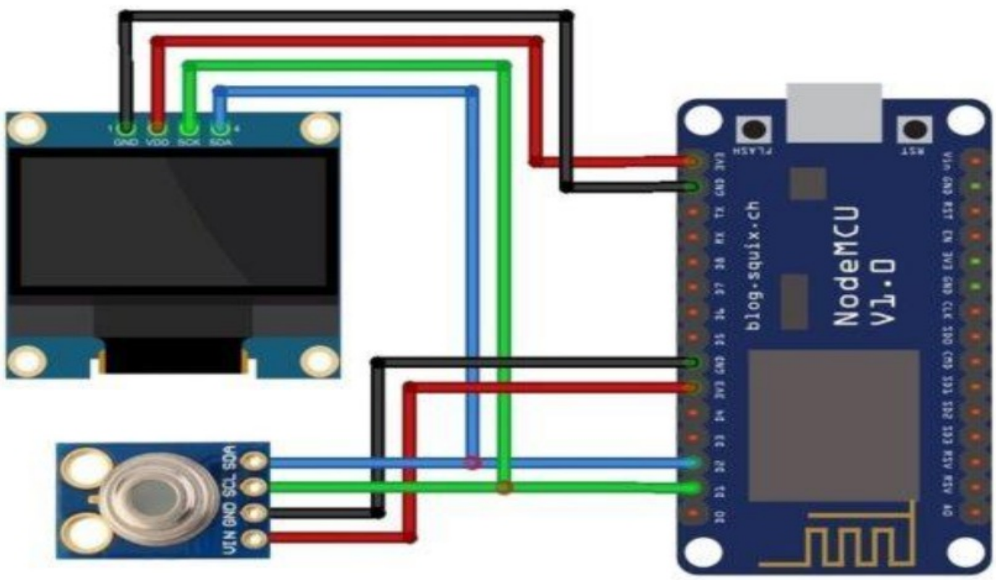


Fig 2.3 Arduino UNO and Nodemcu interfacing with MLX90614



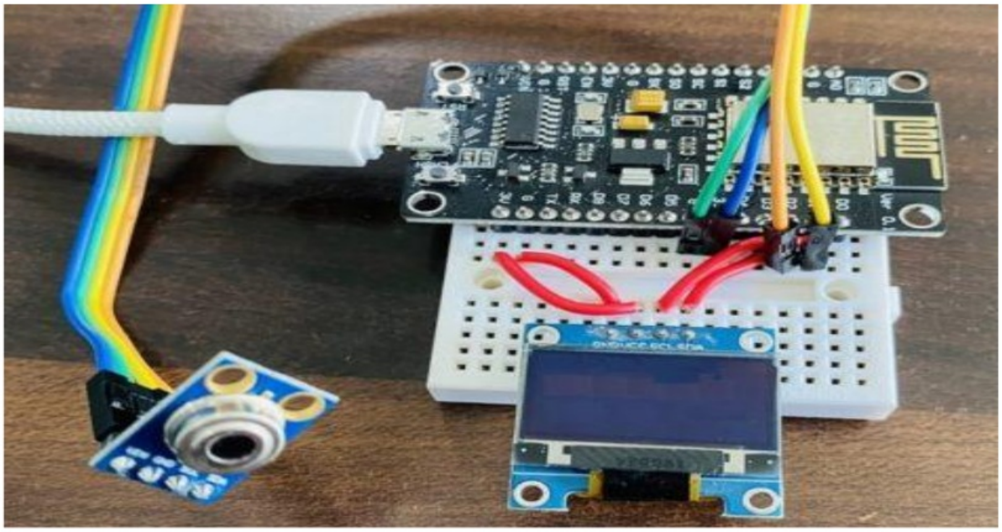


Fig 2.4 Connection diagram of MLX90614 with nodemcu board

SDA	D2
SCL	D1
VCC	3.3V
GND	GND

Table 3.2

2.7 Board Schematic representation :-

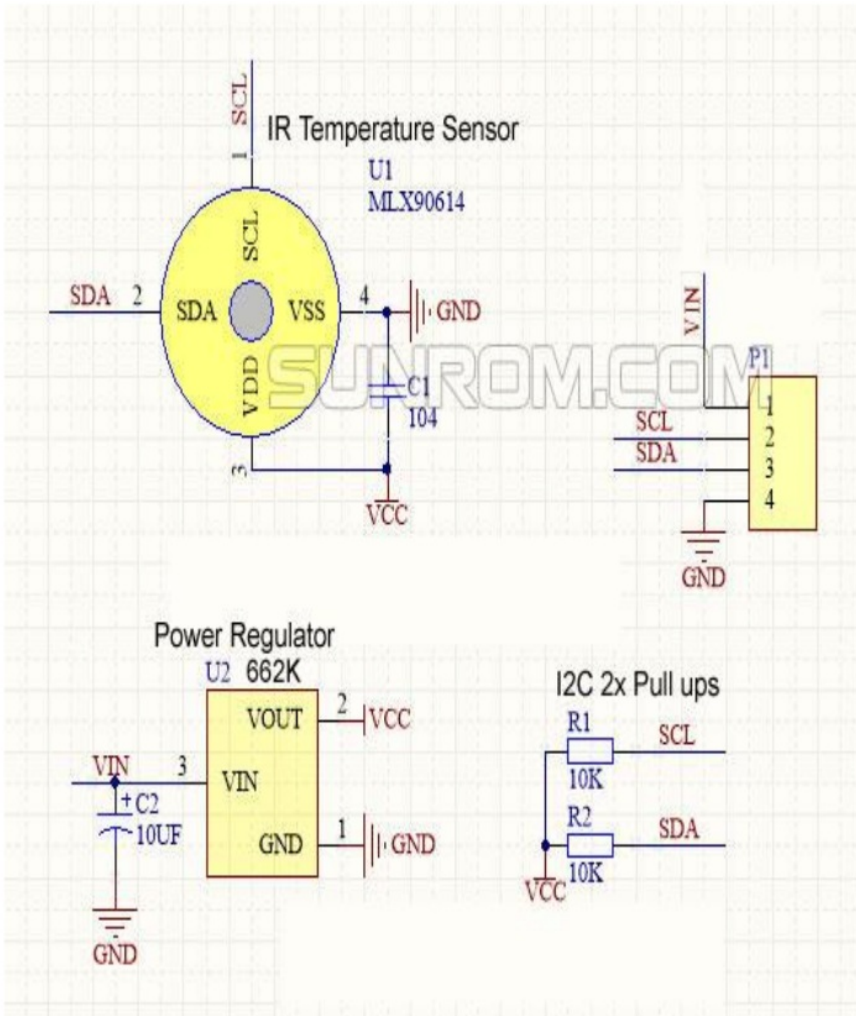


Fig 2.5 - Board schematic representation

2.8 MAX 30102 Pulse Rate Sensor:-

Product Description:-

The MAX30102 is an integrated pulse oximeter and heart rate monitor bio sensor module. It integrates a red LED and an infrared LED, photo detector, optical components, and low-noise electronic circuitry with ambient light suppression. - The MAX30102 features a 1.8V power supply and a separate 5.0V power supply for internal LEDs for heart rate and blood oxygen acquisition in wearable devices, worn on the fingers, earlobe, and wrist. The standard I2C-compatible communication interface can transmit the collected values to the KL25Z and other micro controllers for heart rate and blood oxygen calculation. In addition, the chip can also shut down the module through software, the standby current is close to zero, and the power supply is always maintained. Because of its excellent performance, the chip is widely used in the Samsung Galaxy S series mobile phones. Compared with the previous generation MAX30100, the chip integrates a glass cover to effectively eliminate external and internal light interference, and has the best reliable performance. The Main Parameters: LED peak wavelength: 660nm/880nm LED power supply voltage: 3.3~5V Detection signal type: light reflection signal (PPG)Output signal interface : I2C interface-communication` interface voltage: 1.8~3.3V~5V (optional)Board reserved assembly hole size: 0.5 x 8.5mm Pin Description: VIN: main power input terminal 1.8-5V3-bit pad: Select the pull-up level of the bus, depending on the pin master voltage, select 1.8v or 3_3v (this terminal contains 3.3V and above)SCL: the clock connected to the I2C bus;SDA: data connected to the I2C bus;INT : Interrupt pin of the MAX30102 chip;RD: RED LED ground terminal of MAX30102 chip, generally not connected;IRD: The IR LED ground of the MAX30102 chip is generally not connected;GND: Ground wire.

Fig 2.6 MAX 30102 pulse rate sensor

2.9 Detailed Pin description :-

Pin No.	Pin Name	Description
1	Vin	Main power supply input 1.8v-5v
2	GND	Ground Wire
3	SDA – Serial Data	Data connected to the I2C bus.
4	SCL – Serial Clock	The clock connected to the I2C bus.

Table 3.3

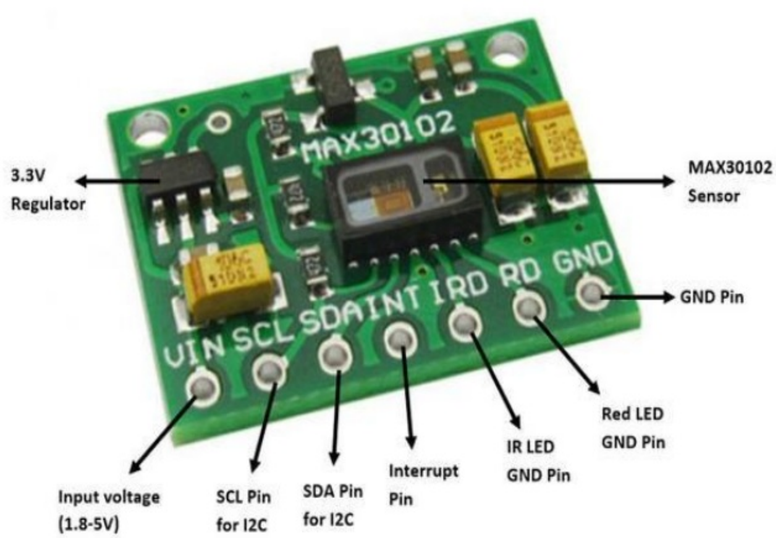


Fig 2.7 MAX 30102 Pin out configuration

2.10 Advantages of MAX3102 in compare to other pulse rate sensor:-

- The MAX30102 integrates red and IR LEDs to modulate LED pulses for oxygen saturation (SpO_2) and heart rate measurements.

- Space savings: Maintains a very small solution size without sacrificing optical or electrical performance; Integrates internal LEDs, photodetectors, optical elements, and low-noise electronics with ambient light rejection.

2.11 Specifications:

- Highly-integrated, small-size sensor
- Non-chest based heart-rate/SpO_2 detection
- Ultra-low power consumption
- LED Peak Wavelength: 660nm/880nm
- LED Supply Voltage: 3.3~5V
- Interface: I2C Interface
- Operating Temperature Range: -40°C to +85°C
- Dimension: 20.3 x 15.2mm
- Weight: 1.1g

2.11 Principle description: -

Photo-dissolution method:

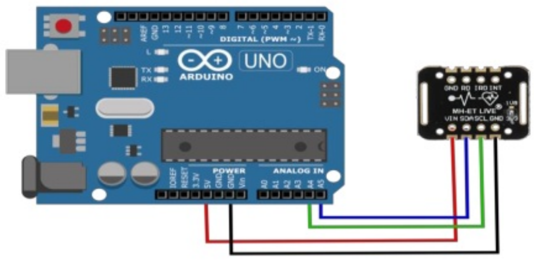
The measurement of pulse and blood oxygen saturation is performed by using human tissue to cause different light transmittance when the blood vessel beats.

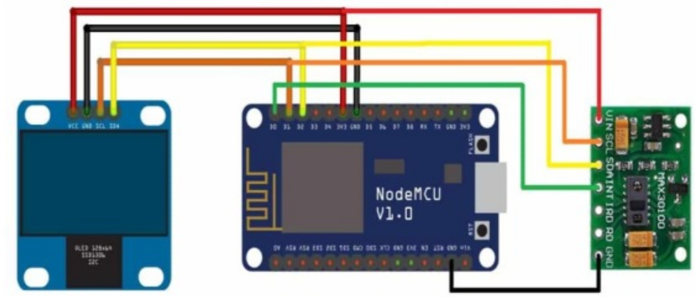
Light source: A specific wavelength of light-emitting diode selective for oxyhaemoglobin (HbO_2) and haemoglobin (Hb) in arterial blood.

Light transmittance is converted into an electrical signal:

The change in the volume of the arterial pulsation causes the light transmittance of the light to change. At this time the light reflected by the human tissue is received by the photoelectric transducer, converted into an electrical signal, and amplified and output.

2.12 Arduino UNO and Nodemcu Interfacing:-





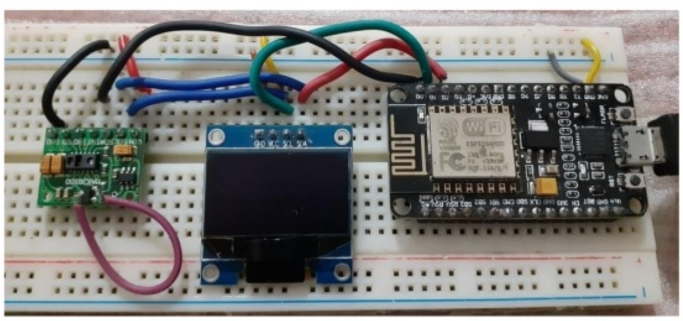


Fig 2.8 Connection diagram of MAX30102 with Nodemcu

SDA	D2
SCL	D1
VCC	3.3V
GND	GND

Table 3.4

2.13 -

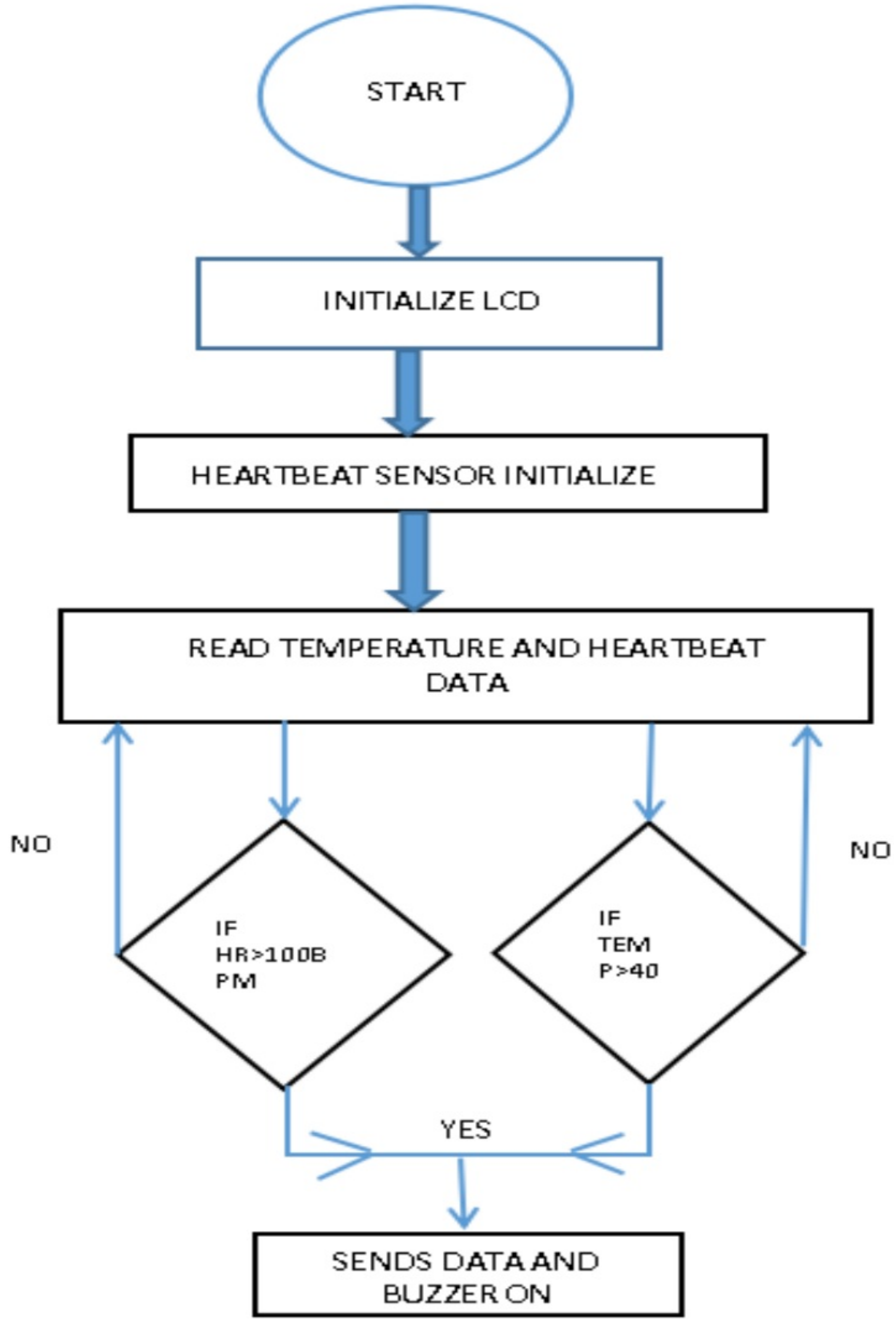


Fig 2.9 Flow diagram (For monitoring parameters)

CHAPTER 3

EMBEDDED SYSTEMS : FALL DETECTION SYSTEM

3.1 Objective (II) : Development of a Wearable-Sensor-Based Fall Detection System for the aged people.

3.1.1 Components Required:-

1. MPU6050 sensor board
2. Jumper wires.
3. Nodemcu ESP8266 board.

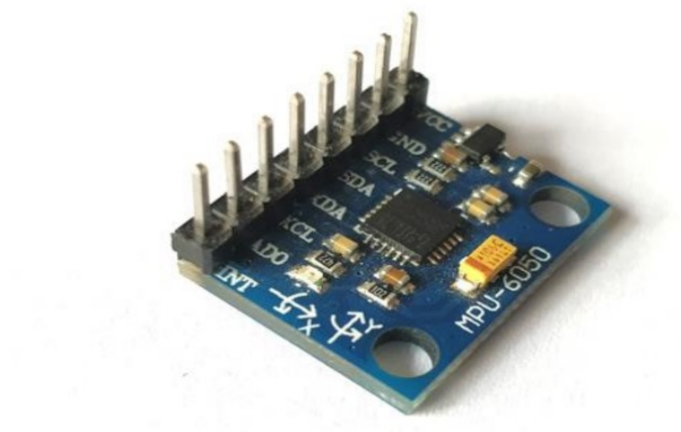


Fig 3.1 MPU6050 - Accelerometer and Gyroscope Module

3.2 Sensor Description :-

The **MPU6050** devices combine a 3-axis gyroscope and a 3-axis accelerometer on the same silicon together with an onboard Digital Motion Processor (DMP) capable of processing complex 9-axis Motion Fusion algorithms.. This helps us to measure acceleration, velocity, orientation, displacement and many other motion related parameter of a system or object. This module also has a (DMP) Digital Motion Processor inside it which is powerful enough to perform complex calculation and thus free up the work for microcontroller.

The module also has two auxiliary pins which can be used to interface external IIC modules like a magnetometer, however it is optional. Since the IIC address of the module is configurable more than one MPU6050 sensor can be interfaced to a Microcontroller using the AD_0 pin. This module also has well documented and revised libraries available hence it's very easy to use with famous platforms like Arduino and ESP8266. So, if we are looking for a sensor to control motion for your RC Car, Drone, Self balancing Robot, Humanoid, Biped or something like that then this sensor might be the right choice.

3.3 Pin Description:-

Table 4.1

Pin No.	Pin Name	Description
1	V_{cc}	Provides power for the module, can be +3V to +5V. Typically +5V is used
2	Ground	Connected to Ground
3	Serial Clock (SCL)	Used for providing clock pulse for I2C Communication
4	Serial Data (SDA)	Used for transferring Data through I2C communication
5	Auxiliary Serial Data (XDA)	Can be used to interface other I2C modules with MPU6050. It is optional

6	**Auxiliary Serial Clock (XCL)**	Can be used to interface other I2C modules with MPU6050. It is optional
7	**AD0**	If more than one MPU6050 is used a single MCU, then this pin can be used to vary the address
8	**Interrupt (INT)**	Interrupt pin to indicate that data is available for MCU to read

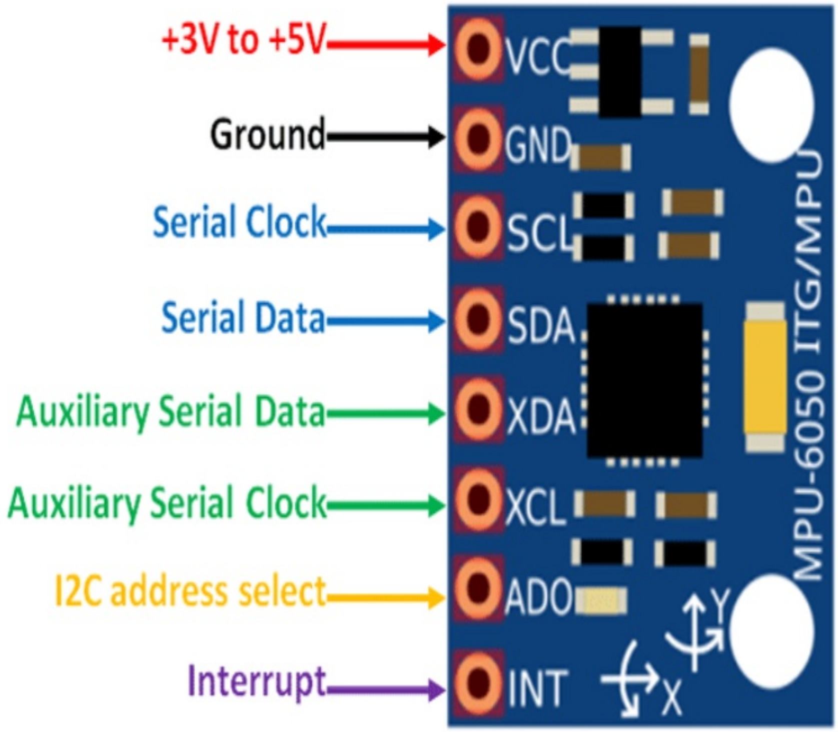


Fig 3.2 MPU6050 PINOUT

3.4 Using procedure of MPU6050:-

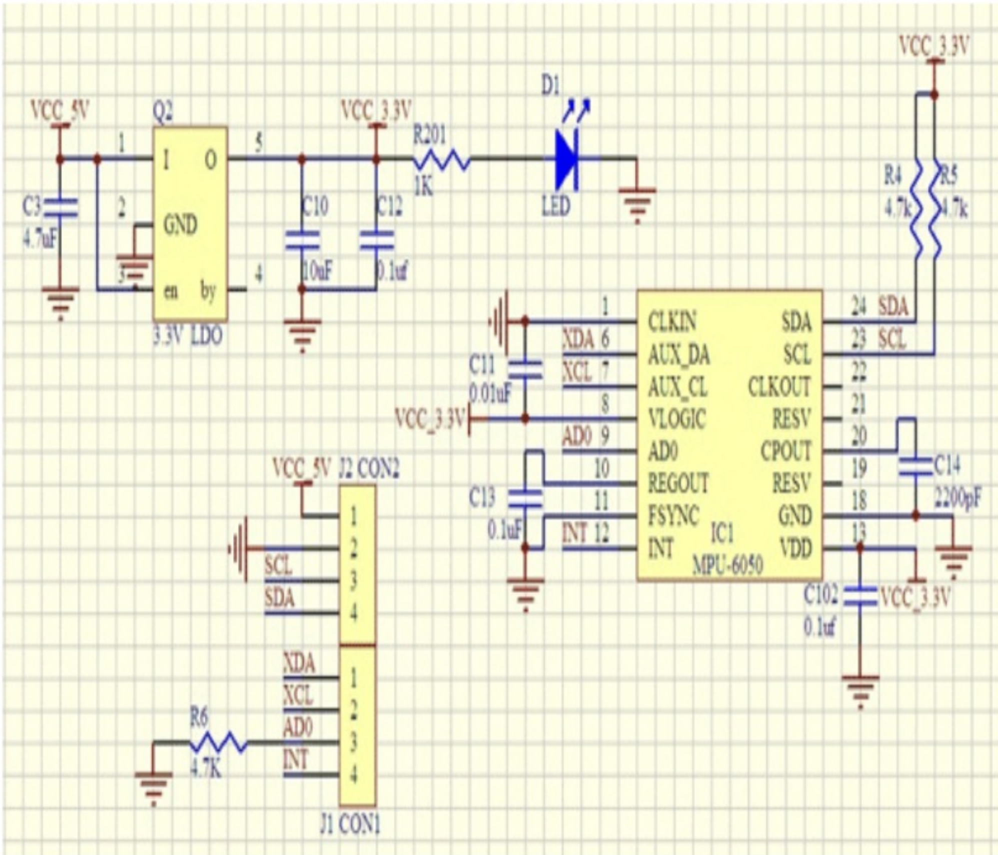


Fig 3.3 Board Schematic

The hardware of the module is very simple, it actually comprises of the MPU6050 as the main components as shown above. Since the module works on 3.3V, a voltage regulator is also used. The IIC lines are pulled high using a 4.7k resistor and the interrupt pin is pulled down using another 4.7k resistor.

The MPU6050 module allows us to read data from it through the IIC bus. Any change in motion will be reflected on the mechanical system which will in turn vary the voltage. Then the IC has a 16-bit ADC which it uses to accurately read these changes in voltage and stores it in the FIFO buffer and makes the INT (interrupt) pin to go high. This means that the data is ready to be read, so we use a MCU to read the data from this FIFO buffer through IIC communication. As easy as it might sound, you may face some problem while actually trying to make sense of the data. However there are lots of platforms like Arduino using which you can start using this module in no time by utilizing the readily available libraries .

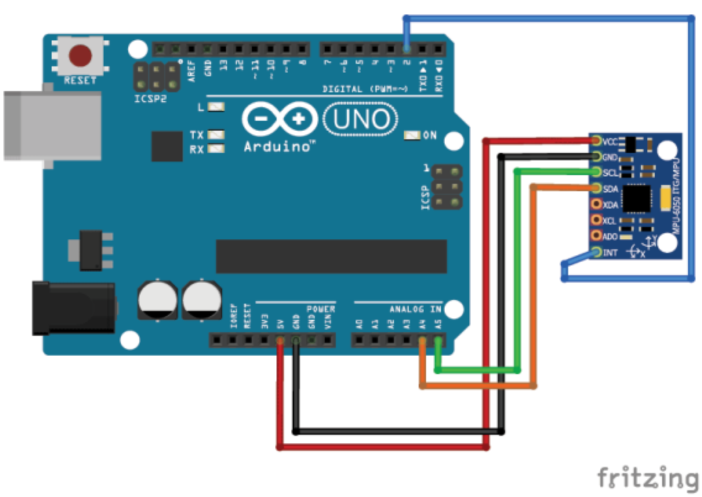
3.5 MPU6050 Features:-

- MEMS 3-aixs accelerometer and 3-axis gyroscope values combined
- Power Supply: 3-5V
- Communication: I2C protocol
- Built-in 16-bit ADC provides high accuracy
- Built-in DMP provides high computational power
- Can be used to interface with other IIC devices like magnetometer
- Configurable IIC Address
- In-built Temperature sensor

3.6 Interfacing MPU6050 with Arduino UNO and Nodemcu board:-

It is very easy to interface the MPU6050 with Arduino due to the library developed by Jeff Row-berg.

Once this library has been added to Arduino IDE, follow the below schematics to establish an IIC connection between your Arduino and MPU6050



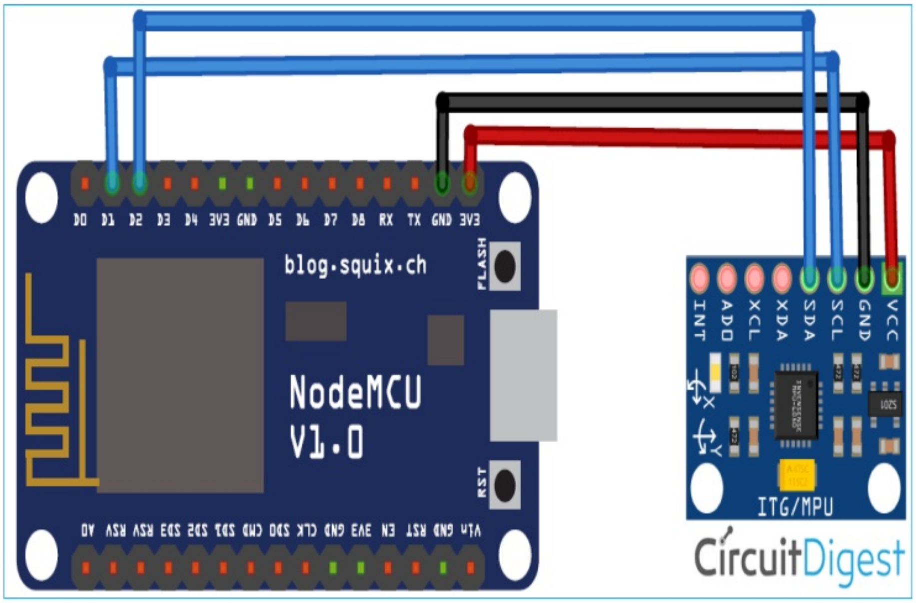


Fig 3.4 Circuit Connection of MPU6050

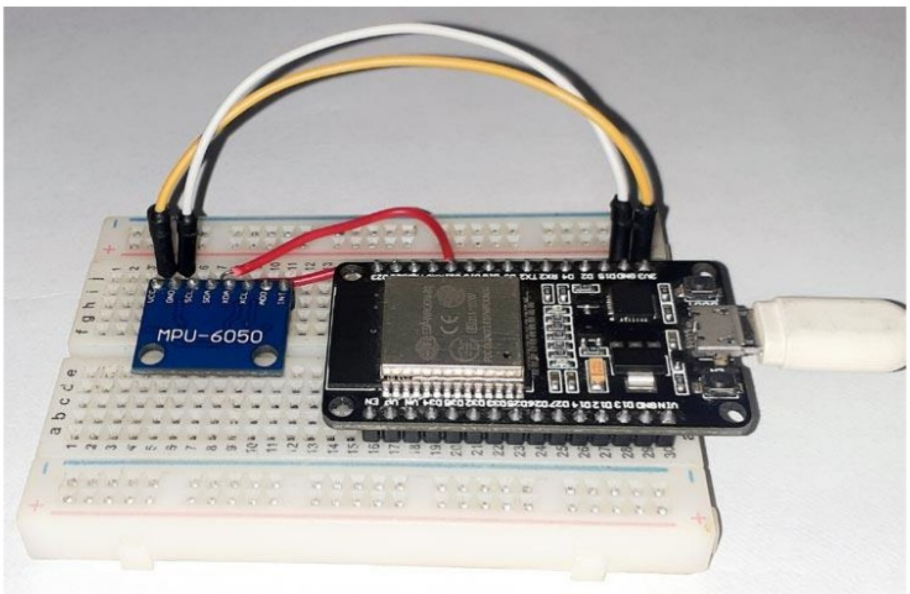


Fig 3.5

3.7 MODEL ACTIVITY

Based on the fact that the direction of gravity is invariably perpendicular to the ground, and that the orientation of the vest worn on the body is supposed to be the same as that of the trunk, we use a Cartesian coordinate system OXYZ for the upper trunk, the origin of which is close to the neck of the human body, and is parallel with the geodetic coordinate system OXYZ, as shown in Figure 1.

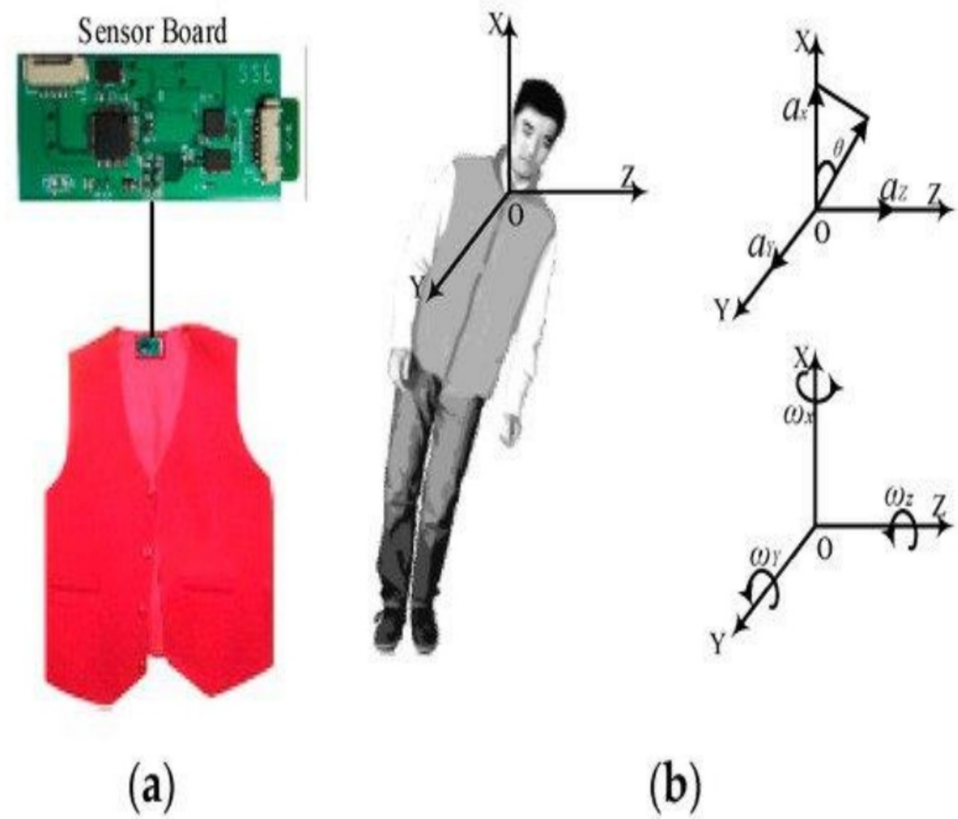


Fig 3.6

Fig 3.6(a) The placement of the sensor board
3.6(b). The geodetic coordinate OXYZ

At a time t, accelerations along the X, Y, and Z axes are denoted as Ax(t), Ay(t) and Az(t) respectively, namely A(t) = {Ax(t), Ay(t), Az(t)}. The resultant acceleration α(t) can be calculated using Equation (1):

$$A(t)= \sqrt{Ax(t)^2 + Ay(t)^2 + Az(t)^2}$$

Since Ax(t), Ay(t) and Az(t) contain an approximation of the gravitational component of the acceleration on every axis, the trunk angle (namely θ(t)) can be calculated using Equation (2):

$$\theta(t)= \cos^{-1}\left(\frac{Ax(t)}{\sqrt{Ax(t)^2 + Ay(t)^2 + Az(t)^2}}\right)$$

The x-axis is perpendicular to the gravitational direction in the lying position and parallel to the gravitational direction in the standing position. A fall usually means that the trunk changes from a standing position to a lying position, and the θ(t) increases from about 0° to about 90°.

Meanwhile, the tri-axial angular velocities of the trunk can be collected by gyroscope. Wx(t), Wy(t), Wz(t) are the angular velocity at time t in the X, Y, and Z axes respectively, namely W(t) = {Wx(t), Wy(t), Wz(t)}. The resultant angular velocity W(t) can be calculated using Equation (3):

$$W(t) = \sqrt{Wx(t)^2 + Wy(t)^2 + Wz(t)^2}$$

3.8 Data Acquisition :-

Fig 3.7 shows the sensor board, which is about 65 mm × 40 mm × 7 mm (length × width × thickness), and is appropriate for use in a vest. The sensor board contains a low-power microcontroller, and a class 2 Bluetooth module. The range of the Bluetooth module is 10 m, and its default transmission rate is 115,200 bps. The range of the tri-axial accelerometer is ±16 g. The full-scale range of the tri-axial gyroscope is ±2000°/s. The sampling data from the tri-axial accelerometer and gyroscope are read and transmitted to an Android smart phone.

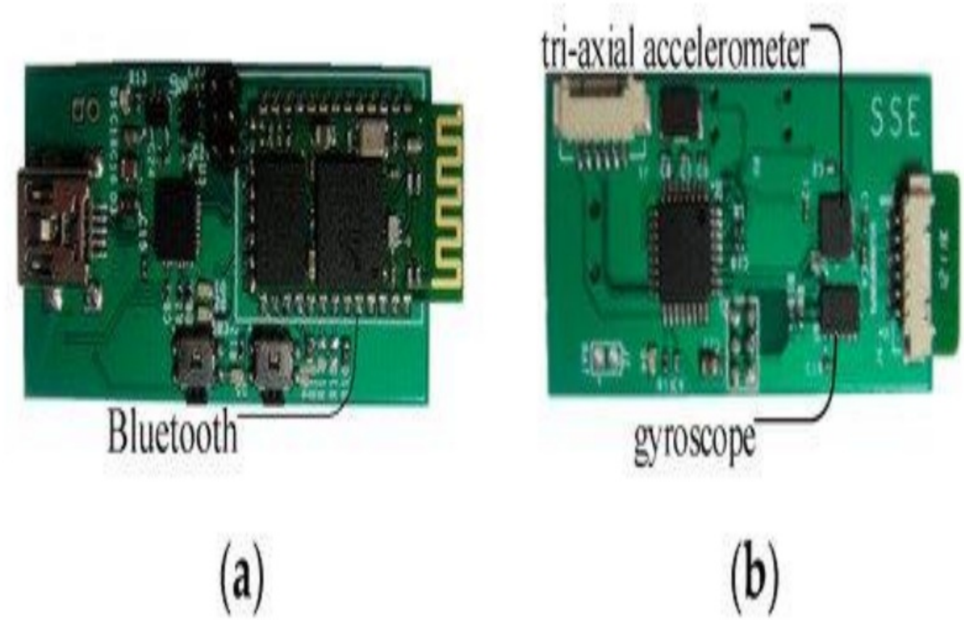


Fig 3.7(a & b) Front & back view of sensor board with Bluetooth, tri-axial accelerometer and gyroscope

Since most frequencies of human activities are less than 20 Hz, the sampling frequency from human activities is set to 100 Hz. The sensor board can acquire tri-axial accelerations and angular velocities, and send them directly to an Android smart phone.

Since falls are usually characterized by rapid acceleration and great angular velocity, four typical subcategories of ADLs and two kinds of falls are proposed in order to find out the difference between ADLs and falls. ADLs include Walking (Wk), Sitting down (Sd), Squatting down (Sq) and Bowing (Bw). Falls include Sideward fall (Sw-

Fall) and Backward fall (Bw-Fall). Twenty healthy individuals, including ten males and ten females, aged 20–45 years were asked to do the simulated falls and normal ADLs, both outdoors and indoors. Each individual performs each of the six kinds of ADLs and falls—Wk, Sd, Sq, Bw, Sw-Fall and Bw-Fall—five times. Hence, the total experimental data set numbers 600 elements, which consists of six 100-element sets for Bw-Fall, Sd-Fall, Wk, Sd, Sq and Bw, respectively, with each one having the same sample length

3.9 Flow Diagram :-

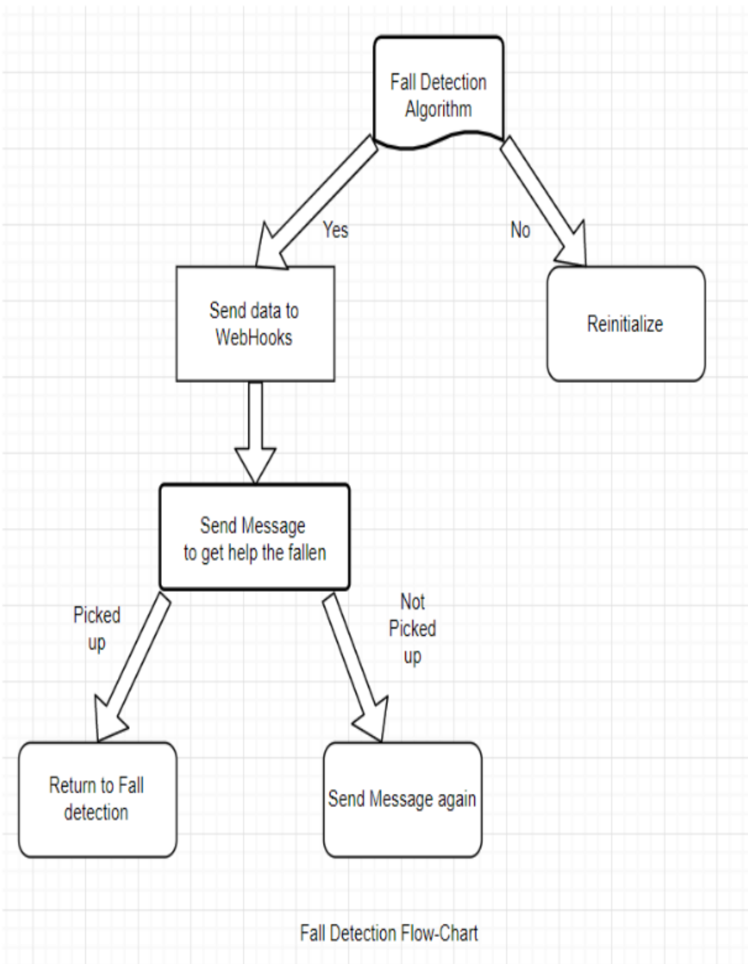


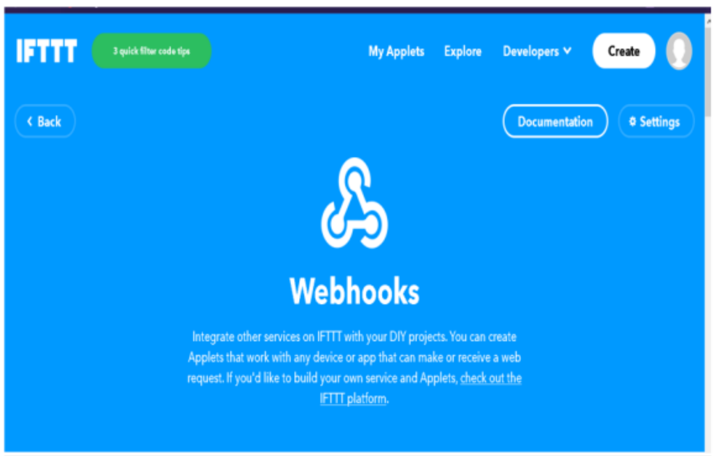
Fig 3.8 Flow diagram

3.10 IFTTT Integration:-

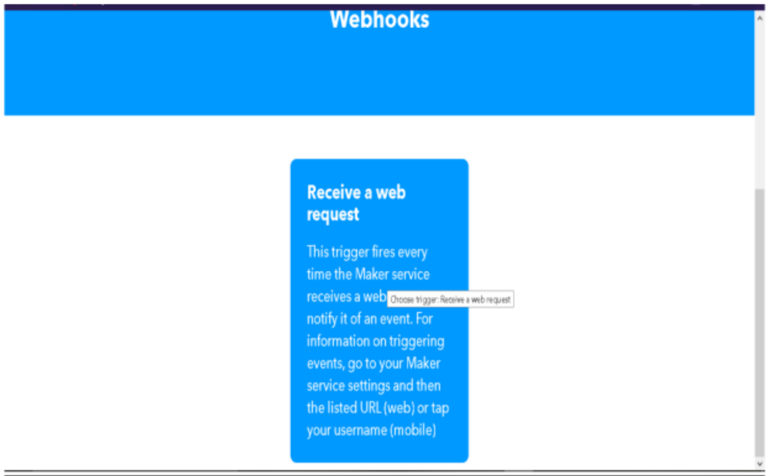
IFTTT (**If This Then That**) is a web-based service by which we can create chains of conditional statements, called applets. Using these applets, we can send Emails, posts to Twitter, posts to Facebook, play music, SMS, notifications, etc. Here in this project, we are using IFTTT to send SMS notifications to the mobile phone when the system detects a fall.

To use the IFTTT, sign in to your IFTTT account if you already have one or create an account. Following are the steps:

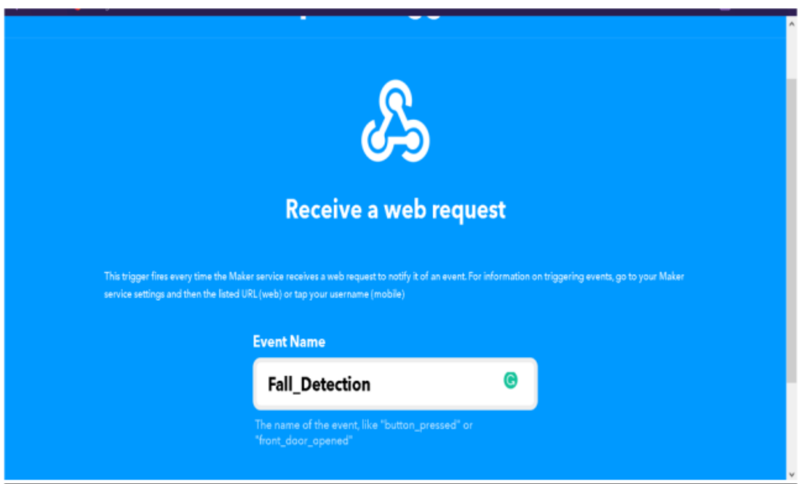
- Click on the documentation to get the key which will be used in programming part



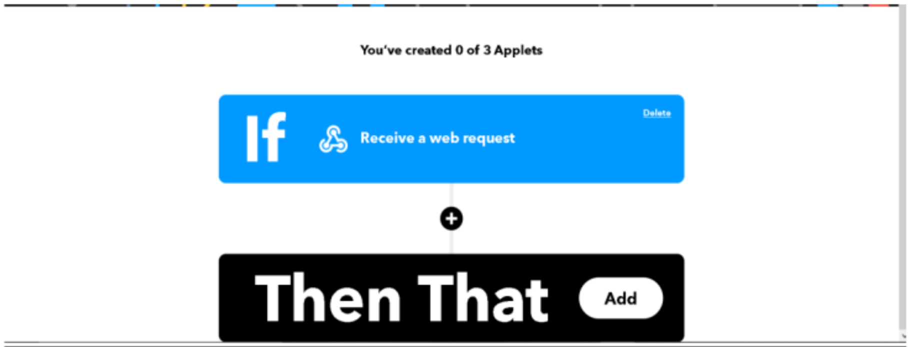
- After that click on create
- Then click on "this"
- Next, search for webhooks and select it



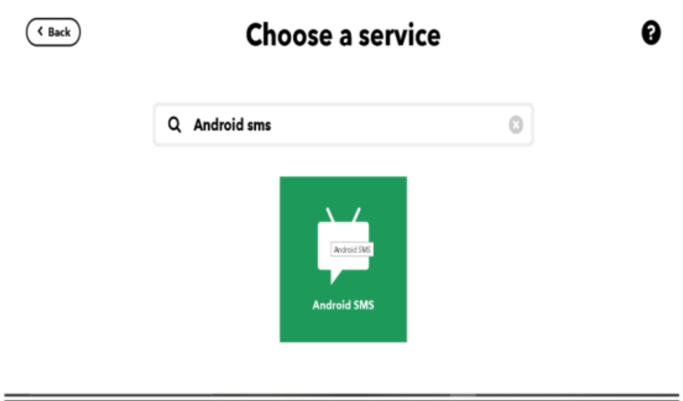
- After clicking the webhooks, now you want to select "Receive a web request" and give an event name as per your wish. In this case, fall detect given



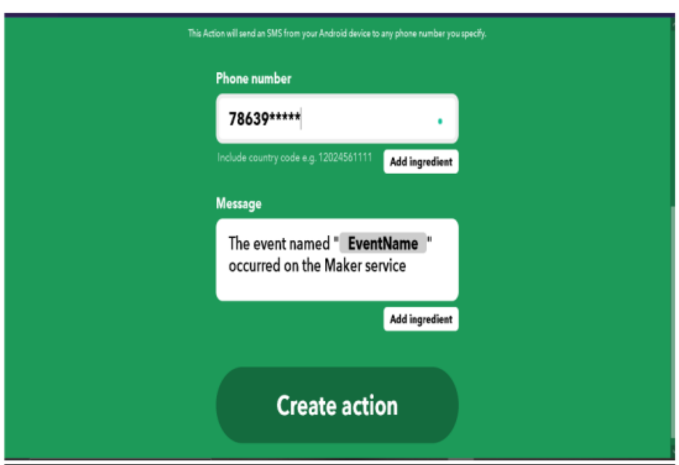
- Then click on "that"



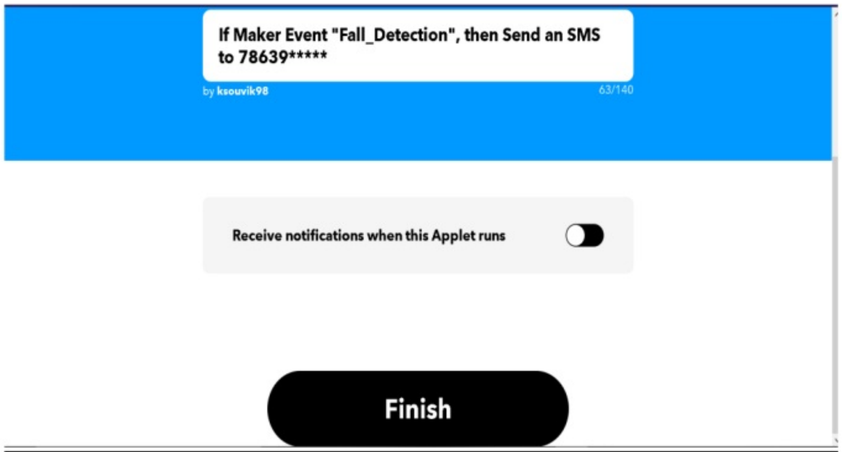
- Search for Android SMS



- Now, you want to enter your phone number with country code



- Then click on create action



- After that click on Create Action.

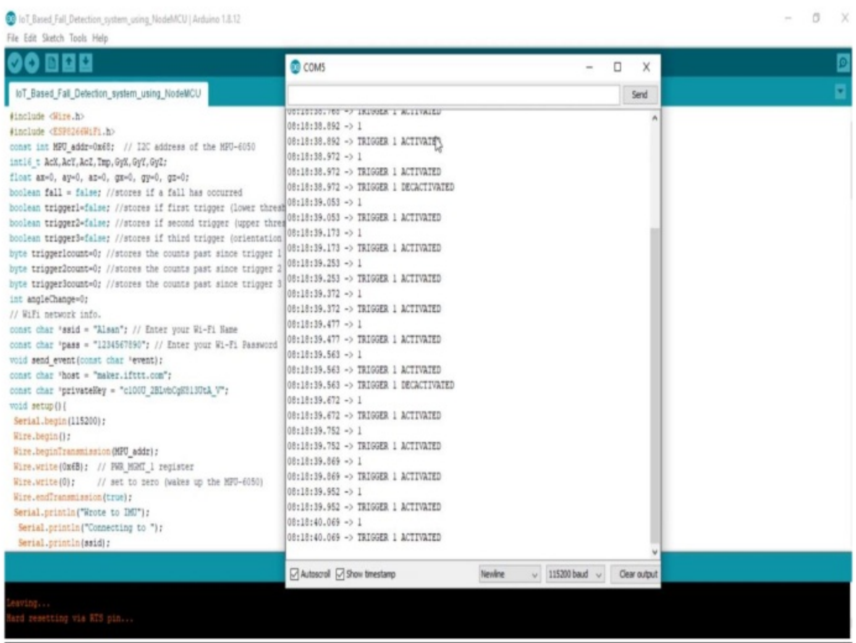


Fig 3.9 Illustration of IFTTT steps

Fig 3.10 Connection diagram

CHAPTER-4

IOT- WEBSITE MONITORING

Objective:- To effectively monitor the patient parameters via a website.

IDEATION:-The parameters that are being monitored by the sensors are checked via a website that will display the sensor data in a graphical way for better understanding and to keep check on the threshold limits of the parameters.

4.1 Introduction to Thinger.io:-

Thinger.io is a cloud IoT Platform that provides every needed tool to prototype, scale and manage connected products in a very simple way. The goal is to democratize the use of IoT making it accessible to the whole world, and streamlining the development of big IoT projects.

- **Free IoT platform**: Thinger.io provides a lifetime freemium account with only few limitations to start learning and prototyping when our product becomes ready to scale, we can deploy a Premium Server with full capacities within minutes.
- **Simple but Powerful**: Just a couple code lines to connect a device and we can start retrieving data or controlling its functionalities with the web-based Console, able to connect and manage thousands of devices in a simple way.
- **Hardware agnostic:** Any device from any manufacturer can be easily integrated with Thinger.io's infrastructure.
- **Extremely scalable & efficient infrastructure:** due to the unique communication paradigm, in which the IoT **server subscribes device resources** to retrieve data only when it is necessary, a single Thinger.io instance is able to manage thousands of IoT devices with low computational load, bandwidth and latencies.

- **Open-Source**: most of the platform modules, libraries and APP source code are available in the Github repository to be downloaded and modified with MIT license.

Thinger.io is an Open-source cloud-based IoT Platform developed by INTERNET OF THINGER SL, a Spanish company whose objective is to provide an efficient, consistent and easy to use technology for IoT.

Thinger.io Platform project started life in early 2015, despite some of the first source code lines was made on 2014, as a side-project by Ph.D. Alvaro Luis Bustamante when he was working as researcher at the University Carlos III of Madrid (Spain), when he was studying different solutions to work online with cheap electronic devices, and he found that all the existent platforms were very difficult to use, inefficient or doesn't had enough capacities to create IoT projects in a simple way.

4.2 Thinger.io Main Features:-

Thinger.io platform is formed by two main products - a Backend (which is the actual IoT server) and a web-based Frontend that simplifies working with all the features using any computer or smartphone. The image below shows the main features provides by this platform to create IoT projects.

- **Connect devices:** Fully compatible with every kind of device, no matter the processor, the network or the manufacturer. Thinger.io allows to create **bidirectional communications** with Linux, Arduino, Raspberry Pi, or MQTT devices and even with edge technologies like Sigfox or LoRaWAN or other internet API data resources.
- **Store Device Data:** Just a couple clicks to create a Data Bucket a store IoT data in a scalable, efficient and affordable way, that also allows real-time data aggregation.
- **Display Real-time** or **Stored Data** in multiple widgets such as time series, donut charts, gauges, or even custom made representations to create awesome dashboards within minutes.

- **Trigger events and data values** using an embedded Node-RED rule engine
- **Extend with custom features** with multiple plugins to integrate IoT projects into your company's software or any other third party Internet service.
- **Custom the appearance** thanks to the fully rebrand able frontend, that allows introducing our branding colours, logotypes and web domain

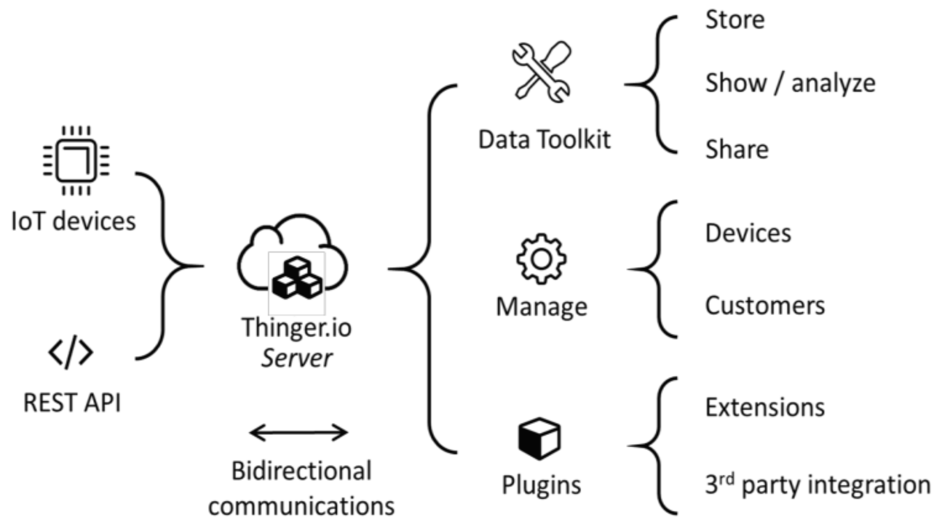


Fig 4.1 Connectivity of thinger.io platform

The core of Thinger.io IoT Platform is designed to be lean and lightweight, to maximize flexibility and minimize learning and common configuration of IoT networks. Plugins offer custom functions that can be deployed on-demand, allowing each user to complement their IoT server as its specific needs.

Dashboards are a data visualization tool that allow all users to understand the analytics that matter to their business, department or project. Even for non-technical users, **dashboards** allow them to participate and understand the analytics process by compiling data and visualizing trends and occurrences. We plan to make a web based dashboard for effective and easy monitoring and understanding of the various health parameters of the patient, such as blood pressure and temperature.

Thinger.io dashboard system is a feature that allows creating nice data representation interfaces within minutes in a very simple way. No coding is required, just selecting different widgets from a list and using drag drop technology to configure the layout of the dashboard, then using the configuration forms it is possible to set the data sources, sampling interval, and other behaviours of each widget. The main types of these widgets are:

- **Real-time** data representation
- **Historical** data representation from buckets
- **Control** device functions or change values with On/Off buttons or sliders

Why this approach?

We could have used normal website coding (using HTML,CSS) to design a website for displaying the parameters. But we preferred using this since we found that it was open source,easy to use and most importantly, it was convenient to create simple,attractive dashboards using this platform. Moreover, the array of devices supported by thinger.io is humongous. So, a wide variety of devices can be easily integrated in the project, irrespective of the manufacturers or properties.

4.3 Working Procedure:-

1. Create device – create a device on thinger.io by registering a new device. By logging into console dashboard, go to Devices ->Add Device.

   Add the device credentials such as identifier, description and credentials, like Wi-Fi username, etc. as mentioned in the code.

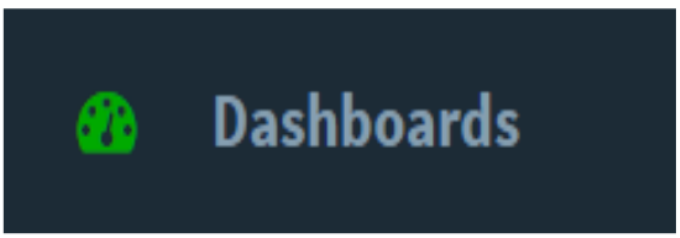
If everything is successful a message appears and the device is displayed in the list of devices.

2. Add a display widget -while editing on the dashboard, a new button Add Widget appears, where the type can be selected based on requirement.

Widget selection is done like this –

Fig 4.2 Illustration of widget selection

3. To manage all our dashboards, it is necessary to access to the Dashboards section, by clicking in the following menu item :



Then click on the Add Dashboard button that will open a new interface for entering the dashboard details.

It is necessary to configure different parameters:

- Dashboard Id: Unique identifier for your dashboard.
- Dashboard name: A representative name of your dashboard, in a more friendly way than its identifier.

- Dashboard description: Fill here any description or detailed information you need to keep about the dashboard.

4. Creating a data bucket – it is a kind of virtual storage for time series info over time. This can be used to plot info in dashboards, or can be exported in different formats for offline processing.To create a data bucket, we need to access Data buckets feature and Add Bucket.

Here are some dashboard layouts created using the platform :-

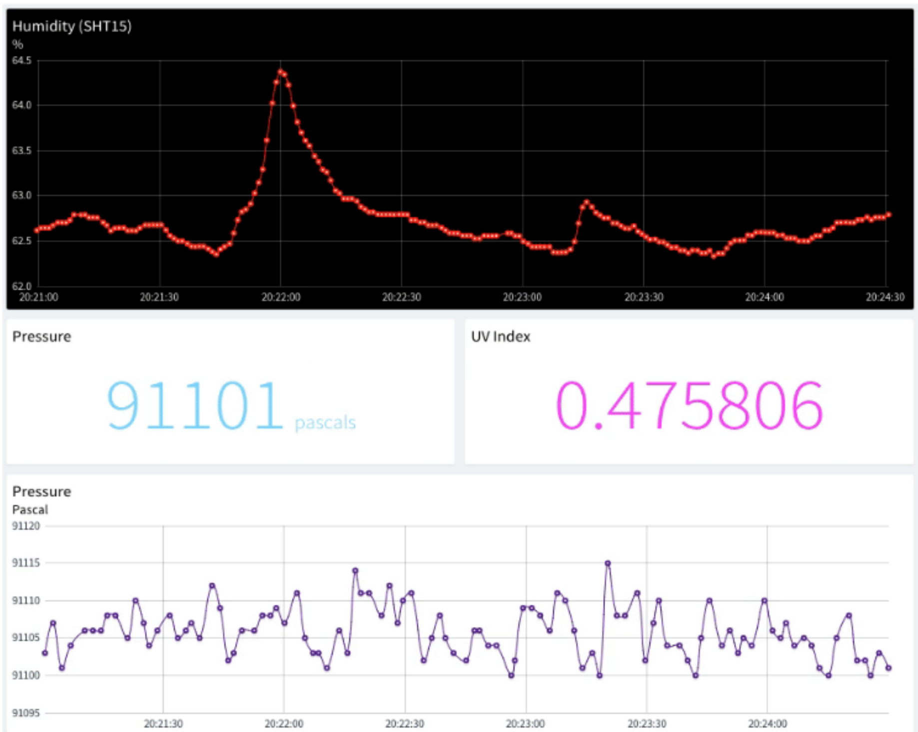


Fig 4.3 Dashboard layout

Here is an example dashboard with some widgets defined, like time series charts, donut charts, maps, or single values.

Once created, dashboards can be shared with third parties through a link or configured as templates to analyse data from different devices of the same type.

The following subsections describe the different parameters for each widget type.

Time Series Chart

A time-series chart is a graph that can display values over time. In this sense, this is quite useful when it is required to display time-series data, like temperature variable that changes over time. It is possible to plot a single variable or multiple values in the same chart.

Donut Chart

A donut chart is a graph that can display a value, normally in form of a rounded percentage. In this sense, this is quite useful when you have a know variable that oscillates between a maximum and minimum value. In this case, it is only possible to only represent a single variable, that can be both updated in real-time from a device, or from a data bucket.

Text/Value

The text/value widget is an useful widget to display any arbitrary data, specially text values that cannot be represented with other widgets. As any other widget, can display data both from connected devices or data buckets.

Clock

This widget is just a clock widget that can display the current time both in the local time zone or in UTC, which can be useful when monitoring processes in real-time. This widget takes the current time just from our computer.

A Dashboard Tab is an additional work page that can be added to a dashboard to organize the visualization of data and simplify navigation between related pannels. The widgets and data sources of each tab can be completely independent of the others but all the tabs will share the same configuration settings (column number, background image, widgets border-radius, etc).

This feature also has the advantage of keeping all the tabs of a dashboard open even if they are not being visualized, so the data of the devices shown in real-time will not be lost when changing from one tab to another.

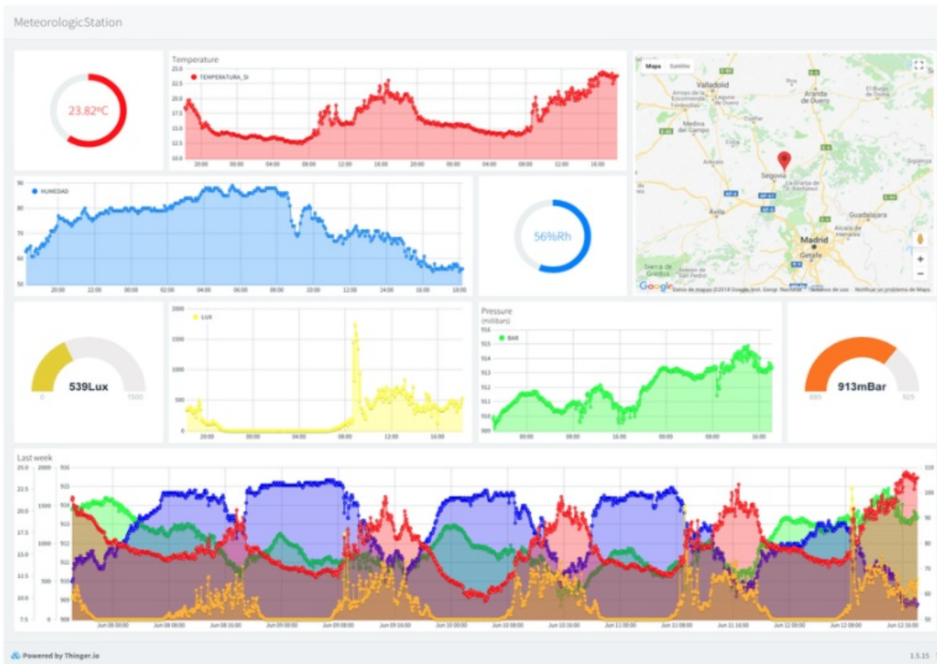
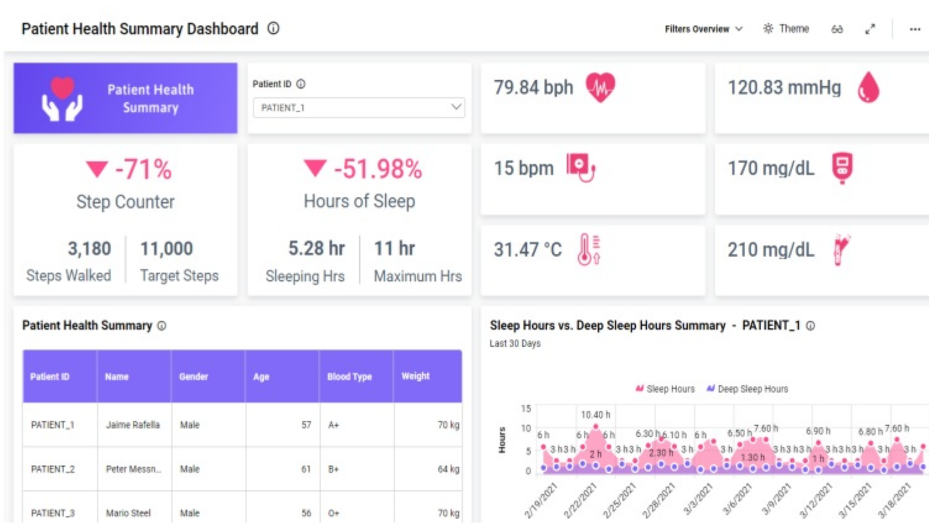


Fig 4.4 (a), (b) An example of how a dashboard monitor looks like



(b)

CHAPTER 5
IOT - BLYNK

OBJECTIVE:- Using Blynk interface to create an user friendly GUI for the patient parameters

IDEATION:-

The parameters that are being monitored by the different sensors are visualized remotely via the Blynk app. This allows a quality GUI experience for the customers i.e. the relatives and family members of the patient. By suing the Blynk app anyone who is shared with the QR code can download the app and it will contains the required project for metrics monitoring. This approach will reduce the amount of panic within family members and will provide a suitable environment and time duration for the best possible.

5.1 About Blynk Community :-

Blynk was originally designed for the Internet of Things. It can control hardware remotely, it can display sensor data, it can store data, visualize it and do many other cool things.

There are three major components in the platform:

- **Blynk App** - allows to you create amazing interfaces for your projects using various widgets we provide.
- **Blynk Server** - responsible for all the communications between the smartphone and hardware. We can use our Blynk Cloud or run our private Blynk server locally. It's open-source, could easily handle thousands of devices and can even be launched on a Raspberry Pi.
- **Blynk Libraries** - for all the popular hardware platforms - enable communication with the server and process all the incoming and outcoming commands.

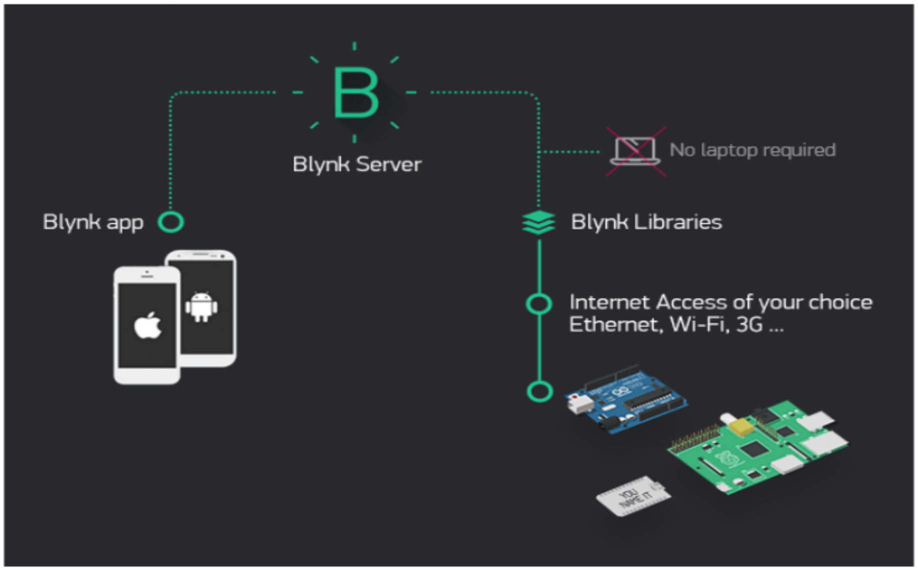


Fig 5.1 Organization of Blynk cloud

5.2 Blynk Features:-

- Similar API & UI for all supported hardware & devices
- Connection to the cloud using:
- WiFi
- Bluetooth and BLE
- Ethernet
- USB (Serial)
- GSM
- Set of easy-to-use Widgets
- Direct pin manipulation with no code writing
- Easy to integrate and add new functionality using virtual pins
- History data monitoring via SuperChart widget
- Device-to-Device communication using Bridge Widget
- Sending emails, tweets, push notifications, etc.

5.3 Introduction to Blynk account and interface:-

1. Create a Blynk Account

After we download the Blynk App, we'll need to create a New Blynk account. This account is separate from the accounts used for the Blynk Forums, in case we already have one. We recommend using a **real** email address because it will simplify things later.

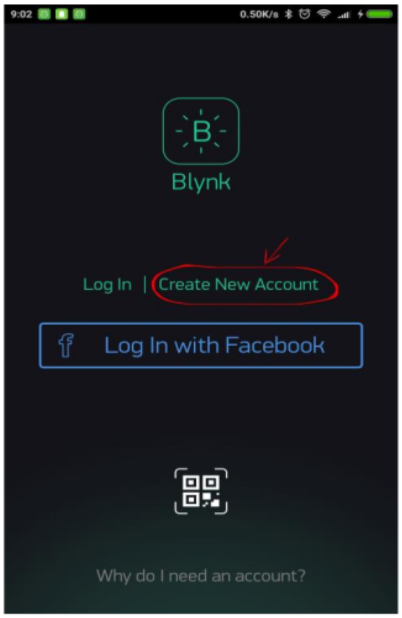


Fig 5.2 Illustration of account creation

2.Create a New Project

After we've successfully logged into our account, we will start by creating a new project.

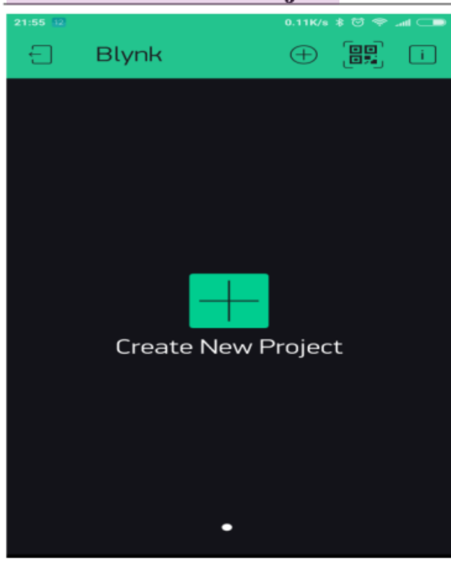


Fig 5.3 Creation of project

3. Choose Your Hardware

Select the hardware model we want to use

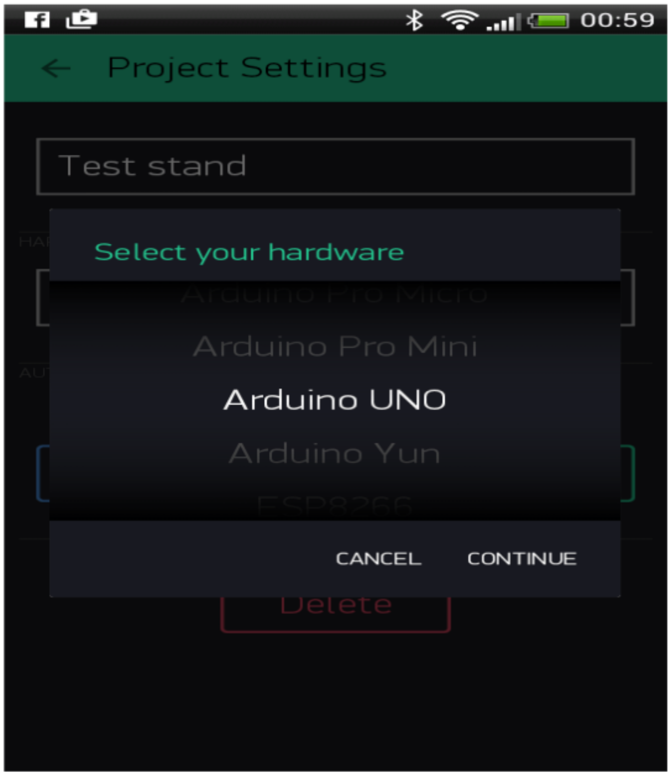


Fig 5.4 Selection of hardware

4. Auth Token

Auth Token is a unique identifier which is needed to connect our hardware to our smartphone. Every new project you create will have its own Auth Token. We'll get Auth Token automatically on our email after project creation. We can also copy it manually. Click on devices section and selected required device :

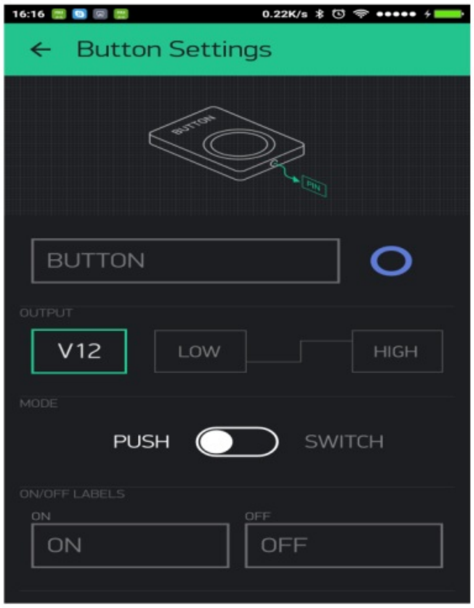
5. Add a Widget

Our project canvas is empty, so we add a button to control our LED.

Tap anywhere on the canvas to open the widget box. All the available widgets are located here. Now pick a suitable button.

Drag-n-Drop - Tap and hold the Widget to drag it to the new position.

Widget Settings - Each Widget has it's own settings. Tap on the widget to get to them.



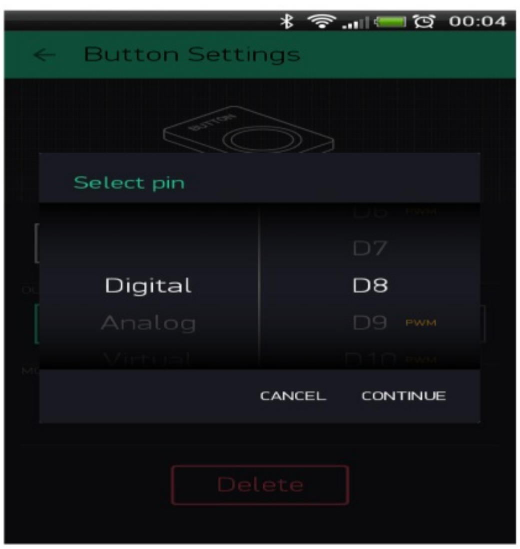


Fig 5.5 Widget selection

6. Run The Project

When we are done with the Settings - we will press the **PLAY** button. This will switch us from EDIT mode to PLAY mode where we can interact with the hardware. While in PLAY mode, we won't be able to drag or set up new widgets, press **STOP** and get back to EDIT mode.We will get a message saying "Arduino UNO is offline

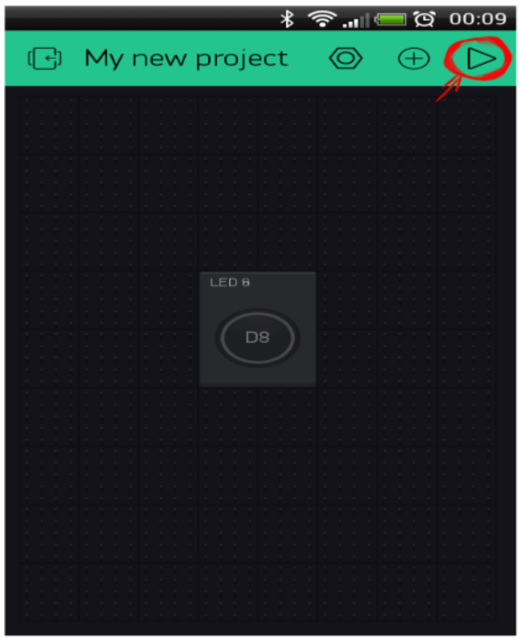


Fig 5.6 Running the project

5.4 Blynk application in metrics monitoring

Through Blynk we can plot the graphical data which will be given by the sensors we will be using.

Blynk uses **virtual pin** concept through which we can push data from the nodemcu to the blynk **mobile app widgets** and also get the data from the widgets to the nodemcu via the ESP8266 Wi-Fi module present.**Virtual Pin** is a concept invented by Blynk Inc. to provide exchange of any data between hardware and Blynk mobile app.

Operation/ Working:-

1. Open Blynk App on mobile and create an account.

2. Create a New Project

3. Select the working board (here NODEMCU)

4. Select the widget as per necessity (here it will be Super chart and Value Display widgets)

5. Tap the widgets for editing its properties. We can use digital or analog or virtual pins as per requirement in the project.

6. After uploading the sensors data in the NODEMCU from Arduino IDE we can monitor the patient parameters on the widgets on the smartphone at a remote distance.

The thing to understand here is every sensor connected to Nodemcu will send its data which we will store in a value and send it to the blynk app via the virtual pin (V1 for eg.)

Let us store the incoming data from the pulse sensor (for eg.) in a variable name : **pulse_val**

Blynk_Write(V1){

Blynk.VirtualWrite(pin,pulse_val);

}

5.5 Blynk Community (My Apps)

- Blynk community has recently launched My Apps for blynk projects
- We need to enter the app name ,icon style ,theme ,color and other attributesfor the app to look smart.
- Then we need to choose the projects we want to access inside the app.
- Blynk community comes up with a exciting features of BlynkFaces which allows us to include multiple projects inside our customised app.
- Then we need to select the type of auth token (Dynamic or static)

- Static auth token is used for devices that connect to the internet over the Etherrnet cable or cellular network.
- Auth token is send to the registered email id in the project itself
- We can publish this app also for others to use and monitor my project readings.

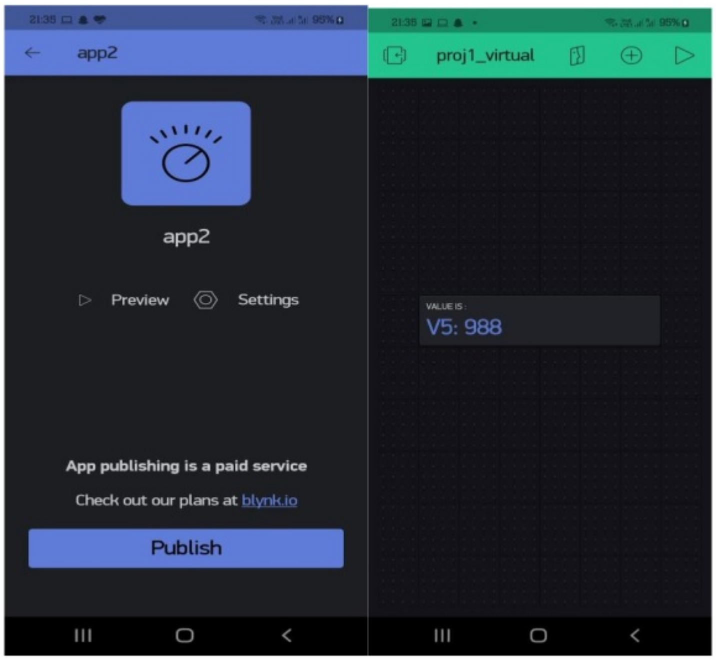


Fig 5.7 Blynk MyApp Layout

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 Summary:-

Health-care is given extreme importance nowadays by each country with the advent of the novel corona-virus. So in this aspect, an IoT based health monitoring system is the best solution for such an epidemic. Internet of Things (IoT) is the new revolution of the internet which is a growing research area, especially in health care.

"Smart IoT based health-care system " is the project where we have mainly focused on two objectives first one was a smart health monitoring system, to collect the health history of patients with a unique ID and store it in a database so that doctors need not spend much of their time in search of the report and give analysis right from the dashboard. Any health-care that is being done will be updated and reflected in the dashboard itself. And the other one is the Wearable-Sensor-Based Fall Detection System for aged people, to monitor the movements of them, and recognize a fall from normal daily activities by using sensors, and automatically sends a request or an alert for help to the caregivers so that they can pick up the patient.

However, there are few shortcomings to this too. A basic knowledge of the operation are to be learnt by the caregivers . Also both the caregiver and the wearer should know how to protect the sensors from water damage or any physical damage .

6.2 Future Scope:-

Even though the healthcare industry was slower in the initial phase to adopt the Internet of Things technologies as compared to other industries but the new sensation Internet of Medical Things (IoMT) is a game-changer in today's world. It is poised to transform how to keep people safe and healthy that too in a cost-efficient way. All these things showing the overall future scope of IoT in the Healthcare domain.

The Internet of Medical Things (IoMT) is an amalgamation of medical devices and mobile applications connecting healthcare information technology systems via various network technologies. The technology can reduce unnecessary hospital visits along with lessening the burden on health care systems by connecting patients to their physicians and doctors while allowing the transfer of medical data that too over a secure network.

As per a report by Frost & Sullivan analysis, the global IoT market was worth $22.5 billion in 2016 and is expected to reach $72.02 billion by 2021, at a compound annual growth rate of 26.2%.

Offering great help to the healthcare domain, the IoMT market is filled with smart devices like wearables, medical/vital monitors for health care use on the body, in the home, community, clinic or hospital settings while associating real-time location, telehealth, and other services.

Some of the ways in IOT in medical services can help in upcoming time are :

- Reducing emergency room waiting time
- Tracking patients, staff, and inventory
- Enhancing drug management
- Enhancing patients' participation while offering them satisfactory result
- Ensuring the availability of critical hardware
- Turning data into actions

(IOMT) benefits include :

ERROR REDUCTION:

The data that generates from the connected devices helps in taking effective and accurate decision making and also ensures smooth operations with a reduction in errors, waste, and costs.

FAST DIAGNOSIS:- Due to continuous patient monitoring and real-time data tracking, doctors can diagnose/detect disease at an early stage or even before the disease actually develops based on symptoms.

COST EFFICIENCY:-By using IoT enabled devices, doctors can monitor patients in real-time. Thus, the process of real-time monitoring at distinct places can help patients cut down not-so-necessary visits to doctors, hospital stays and re-admissions in nearby hospitals.

REMOTE MEDICAL ASSISTANCE:-For lone patients in a medical emergency, it is near next to impossible to contact a doctor who is available miles away. But with IoT application in healthcare and other related devices, it is possible to do so. Not only this, the medics and health personnel can also check the patients to identify the ailments on-the-go.

ALERTS AND TRACKING:-In life-threatening circumstances, on-time alerts become quite critical. To combat such situations, medical IoT devices and applications can gather vital data and transfer it to doctors and health personnel for real-time tracking. Also, these mobile applications and IoT devices can also send notifications regarding a patient's critical conditions irrespective of place, time.

REMOTE REPORTING AND MONITORING: -Through connected devices, it becomes easy for doctors and physicians to monitor patients' health. Also, real-time monitoring can save lives in a medical emergency like diabetic attacks, heart failure, asthma attacks, etc. By means of a smart medical device connected to the smartphone app, collecting medical and other required health data will not be challenging. IoT devices collect and transfer health data like- blood pressure, oxygen, and blood sugar levels, weight, and ECGs. Data collected from these devices are stored in the cloud and can be used by an authorized person, who could be a physician, insurance company, a participating health firm or an external consultant, regardless of their place, time, or device.

As per the latest innovations and requirements in medical technologies, we must form an organized network of smart devices which would collect ,sync and manage huge amount of data effectively and cost efficiently .

In future the scope of automation in health industries is regarded as one of the most revolutionary schemes ever deployed in the healthcare industries. We look forward to this upcoming boost in the infrastructure of the healthcare structure to build a greater and prosperous nation.

6.3 Cost Analysis-

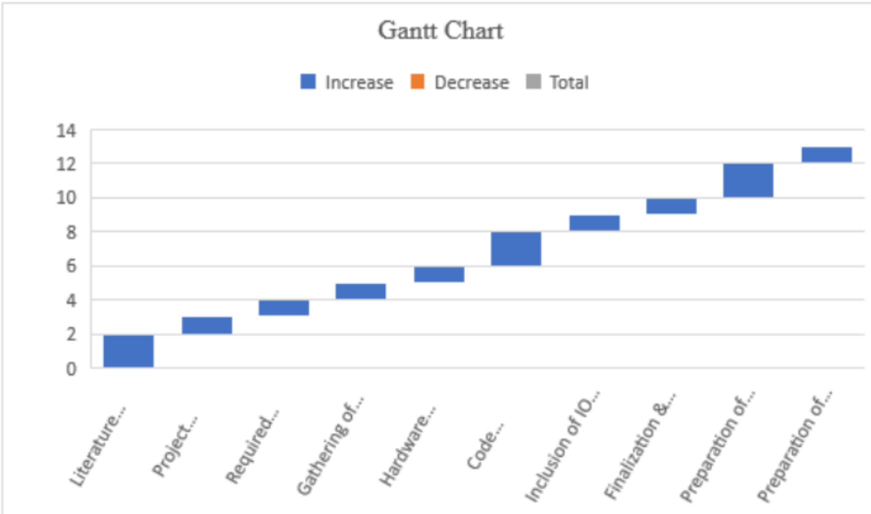
Equipment	**Cost**
MLX 90614 Temperature Sensor.	Rs. 1299
MAX 30102 Pulse Rate Sensor.	Rs. 699
MPU6050 sensor board	Rs. 213
NodeMCU ESP8266 Wi-Fi development board	Rs. 450
Jumper wires	Rs. 199
Total	Rs. 2860

CHAPTER 7
PLANNING & REFERENCES

7.1 Planning and project management

S.No.	Activity	Starting Week	Number of Weeks
1.	Literature Review	1st week of January	2
2.	Project Finalization	3rd week of January	1
3.	Required software setup	4th week of January	1
4.	Gathering of Hardware & Formation of codes	1st week of February	1
5.	Hardware assembly calibration	2nd week of February	1
6.	Code Integration & Debugging	3rd week of February	2
7.	Inclusion of IOT and Blynk	1st week of March	1
8.	Finalization & modification of Website and mobile application	1st week of March	1
9.	Preparation of project report	2^{nd}-3rd week of March	2
10.	Preparation of Project presentation	4th week of March	1

The Gantt Chart is shown below :-



7.2 REFERENCES –

[1]. Md. Milon Islam, Ashikur Rahaman and Md. Rashedul Islam. 2020. Development of Smart Healthcare Monitoring System in IOT Environment *Springer Nature Singapore Pte Ltd*.

[2]. Deepak N, Rajendra Prasad C H and Sanjay Kumar Ch. 2018. Patient Health Monitoring using IOT *International Journal of Innovative Technology and Exploring Engineering* 8.

[3]. Hari Kishore K, Surendra Nath K V, Hari Krishna K V N, Pavan Kumar D, Manikanta V and Fazal Noor Basha. 2019. IOT Based Smart Health Monitoring Alert Device *International Journal of Innovative Technology and Exploring Engineering* 8.

[4]. Tamilselvi V, Sribalaji S, Vigneshwaran P, Vinu P, GeethaRamani J. IoT based health monitoring system. In: 2020 6th International conference on advanced computing and communication systems (ICACCS). IEEE; 2020. p. 386–9

[5]. Riazul Islam SM, Kwak Daehan, Humaun Kabir M, Hossain M, Kwak Kyung-Sup. The Internet of Things for health care: a comprehensive survey. *IEEE Access. 2015;3:678–708*.

[6].Li, Q.; Stankovic, J.A.; Hanson, M.A.; Barth, A.T.; Lach, J.; Zhou, G. Accurate, fast fall detection using gyroscopes and accelerometer derived posture information. In Body Sensor Networks; International Workshop on Wearable & Implantable Body Sensor Networks: Berkeley, CA, USA, 2009; pp. 138–143.

[7].Wang, J., Zhang, Z., Li, B., Lee, S., Sherratt, R.An enhanced fall detection system for elderly person monitoring using consumer home networksIEEE Transactions on Consumer Electronics2014601232910.1109/TCE.2014.67809212-s2.0-84898032338

[8]. Islam M S, Islam M T, Ali F. Almutairi, GankokBeng, Amid N M N. 2019. Monitoring of the
Human Body Signal through the Internet of Things (IoT) Based LoRa Wireless Network
System Journal of Applied Sciences 9

[9]. Desai MR, Toravi S. A smart sensor interface for smart homes and heart beat monitoring using WSN in IoT environment. *In: 2017 International conference on current trends in computer, electrical, electronics and communication (CTCEEC). IEEE; 2017. p. 74–7*

[10]. Prakash H Patil, Pratyush Singh, Swatee Biradar and Prasad Rane. 2013. Design and Implementation of Wireless Patient Health Monitoring System *International Journal of Engineering Research and Technology 2* (6).

[11]. He, J.; Hu, C.; Wang, X.Y. A Smart Device Enabled System for Autonomous Fall Detection and Alert. Int. J. Distrib. Sens. Netw. 2016, 12, 1–10

[12]. Wang, J.; Zhang, Z.; Li, B.; Lee, S.; Sherratt, R.S. An Enhanced Fall Detection System for Elderly Person Monitoring using Consumer Home Networks. *IEEE Trans. Consum. Electron*. **2014**, *60*, 23–28.

[13]. Rahaman A, Islam M, Islam M, Sadi M, Nooruddin S. Developing IoT based smart health monitoring systems: a review. *Rev Intell Artif. 2019;33:435–440. doi: 10.18280/ria.330605.*

[14]. Becker, C.; Schwickert, L.; Mellone, S.; Bagalà, F.; Chiari, L.; Helbostad, J.L.; Zijlstra, W.; Aminian, K.; Bourke, A.; Todd, C.; et al. Proposal for a multiphase fall model based on real-world fall recordings with body-fixed sensors. In *Zeitschrift Für Gerontologie Und Geriatrie; Springer International Publishing: Berlin, Germany*, 2012; pp. 707–715.

[15]. Bourke, A.K.; Van De Ven, P.; Gamble, M.; O'Connor, R.; Murphy, K.; Bogan, E.; McQuade, E.; Finucane, P.; O´Laighin, G.; Nelson, J. Evaluation of waist-mounted tri-axial accelerometer-based fall-detection algorithms during scripted and continuous unscripted activities. *J. Biomech*. **2010**, *43*, 3051–3057.

[16]. Vavoulas, G.; Pediaditis, M.; Spanakis, E.G.; Tsiknakis, M. The MobiFall dataset: An initial evaluation of fall detection algorithms using smartphones. In Proceedings of the IEEE 13th International Conference on Bioinformatics and Bioengineering, Chania, Greece, 10–13 November 2013; pp. 1–4.

[17]. F. Bianchi, S. J. Redmond, M. R. Narayanan, S. Cerutti, and N. H. Lovell, "Barometric pressure and triaxial accelerometery-based falls event detection," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 18, no. 6, pp. 619–627, 2010

[18]. S. D. Bersch, C. M. J. Chislett, D. Azzi, R. Kusainov, and J. S. Briggs, "Activity detection using frequency analysis and off the-shelf devices: fall detection from accelerometer data," in Proceedings of the 5th International Conference on Pervasive Computing Technologies for Healthcare and Workshops (Pervasive Health '11), pp. 362–365, Dublin, Ireland, May 2011.

SELF DECLARATION FOR PLAGIARISM CHECK

We, Souvik Karmakar(1807228), Indrashis Mitra(1807274), Kinjal Sarkar(1807277) and Pratyay Basu(1807291) are declaring that our Project report on "**HEALTHCARE MONITORING AND MANAGEMENT SYSTEM**" has plagiarism well within the limits prescribed to us. We take the full responsibility of it.

HEALTHCARE MONITORING AND MANAGEMENT SYSTEM

ORIGINALITY REPORT

57%	56%	23%	33%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.mdpi.com Internet Source	7%
2	www.appventurez.com Internet Source	7%
3	components101.com Internet Source	6%
4	docs.thinger.io Internet Source	5%
5	www.iieta.org Internet Source	5%
6	icdayroi.com Internet Source	3%
7	jcreview.com Internet Source	3%
8	www.sunrom.com Internet Source	3%
9	osoyoo.com Internet Source	2%

10	theiotprojects.com Internet Source	2%
11	Submitted to Victoria University College Student Paper	2%
12	B Sumathy, S Kavimullai, S Shushmithaa, S Sai Anusha. "Wearable Non-invasive Health Monitoring Device for Elderly using IOT", IOP Conference Series: Materials Science and Engineering, 2021 Publication	2%
13	Submitted to D.K.T.E. Society's Textile and Engineering Institute, Ichalkaranji Student Paper	1%
14	link.springer.com Internet Source	1%
15	kanazawa-u.repo.nii.ac.jp Internet Source	1%
16	Submitted to National Institute of Technology, Rourkela Student Paper	1%
17	www.thetips4you.com Internet Source	1%
18	www.coursehero.com Internet Source	1%

Submitted to Universiti Sains Islam Malaysia

19	Student Paper	1%
20	create.arduino.cc Internet Source	1%
21	www.klipfolio.com Internet Source	1%
22	electronicdesign.com Internet Source	<1%
23	journals.sagepub.com Internet Source	<1%
24	robu.in Internet Source	<1%
25	www.smart-prototyping.com Internet Source	<1%
26	mdpi.com Internet Source	<1%
27	Submitted to De Montfort University Student Paper	<1%
28	research.library.mun.ca Internet Source	<1%
29	docplayer.net Internet Source	<1%
30	Submitted to UCSI University Student Paper	<1%

31	robokits.co.in Internet Source	<1%
32	onlinelibrary.wiley.com Internet Source	<1%
33	www.szapl.com Internet Source	<1%
34	Submitted to University of Sheffield Student Paper	<1%
35	Submitted to Birla Institute of Technology and Science Pilani Student Paper	<1%
36	www.slideshare.net Internet Source	<1%

Exclude quotes	On	Exclude matches	< 10 words
Exclude bibliography	On		