

IMPLEMENTATION OF IOT AND AI BASED SYSTEM TO PROVIDE SMART HEALTHCARE ASSISTANCE

Submitted in partial fulfillment of the requirements for the award of Bachelor of
Engineering Degree in Electronics and Communication Engineering

by

GANGIREDDYGARI CHAKRAPANI REDDY (40130061)

GANTA VENKATA SREEKANTH (40130062)



**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING
SCHOOL OF ELECTRICAL AND ELECTRONICS**

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

CATEGORY -1 UNIVERSITY BY UGC

**Accredited with Grade “A++” by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI – 600119**

APRIL - 2024



SATHYABAMA

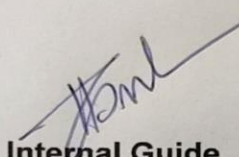
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

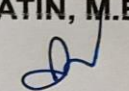
DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **G.CHAKRAPANI REDDY (40130061), G.V.SREEKANTH (40130062)** who carried out the Project entitled "**IMPLEMENTATION OF IOT AND AI BASED SYSTEM TO PROVIDE SMART HEALTHCARE ASSISTANCE**" under our supervision from November 2023 to April 2024.


Internal Guide

Dr. T. BERNATIN, M.E, Ph.D.,


Head of the Department

Dr. T. RAVI, M.E, Ph.D.,

Submitted for Viva voce Examination held on

Internal Examiner

External Examiner

DECLARATION

I, G. CHAKRAPANI REDDY hereby declare that the Project Report entitled "IMPLEMENTATION OF IOT AND AI BASED SYSTEM TO PROVIDE SMART HEALTH CARE ASSISTANCE" done by us under the guidance of Dr. T. BERNATIN M.E, Ph.D., Dept of Electronics and Communication Engineering, SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in ELECTRONICS AND COMMUNICATION ENGINEERING.

DATE: 18-04-2023

PLACE: Chennai

G. Chakrapani Reddy
SIGNATURE OF THE CANDIDATE:

DECLARATION

I, GANTA VENKATA SREEKANTH hereby declare that the Project Report entitled "IMPLEMENTATION OF IOT AND AI BASED SYSTEM TO PROVIDE SMART HEALTH CARE ASSISTANCE" done by us under the guidance of Dr. T. BERNATIN M.E, Ph.D., Dept of Electronics and Communication Engineering, SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY, CHENNAI, is submitted in partial fulfilment of the requirements for the award of Bachelor of Engineering degree in ELECTRONICS AND COMMUNICATION ENGINEERING.

DATE: 15/04/2024

PLACE: Chennai

Gr. V. Sreekanth.
SIGNATURE OF THE CANDIDATE:

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **The Board of Management of Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

I convey my thanks to **Dr. N.M. NANDHITHA M.E., Ph.D., Dean, School of Electrical and Electronics Engineering**, and **Dr. T. RAVI M.E., Ph.D., Head of the Department of Electronics and Communication Engineering** for providing us necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to our Project Guide **Dr. T. BERNATIN M.E, Ph.D.**, dept. of Electronics and Communication Engineering for her valuable guidance, suggestions, and constant encouragement paved way for the successful completion of our project work.

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

ABSTRACT

With the rapid advancement of information and communication technology, IoT and artificial intelligence (AI) have facilitated the emergence of smart healthcare assistance. This integration enables the incorporation of healthcare systems into long-term care institutions. IoT-enabled technologies enable the possibility of developing real time clinical support systems. This project presents an enhanced health care monitoring system. The collected data from sensors, we are using the third-party platforms like Blynk IoT. This will give the delayed output. So, this can be overcome by using the Microsoft Azure cloud which will give faster responses. The integration of Internet of Things (IoT) and Artificial Intelligence (AI) technologies in the healthcare industry will make the advancements in the patient care. This project explores the application of IoT and AI in patient alerting systems, which play a crucial role in providing real-time, pro-active healthcare interventions. AI powered techniques allow healthcare providers to recognizing from normal health conditions and triggering timely alerts when intervention is necessary. This system maintains the balanced health. The paper also highlights key insights for the top application categories, which include wearables and connectivity, disease detection and treatment, patient care, and sensor networks, and identifies gaps and future research directions related to technology design and acceptance, regulations for data security and privacy, and systems efficacy and safety. This project explores the application of IoT and AI in patient alerting systems, which play a crucial role in providing real-time, pro-active healthcare interventions. AI powered techniques allow healthcare providers to recognizing from normal health conditions and triggering timely alerts when intervention is necessary. This system helps to maintain the balanced health.

TABLE OF CONTENTS

CHAPTER No.	TITLE	PAGE No.
	ABSTRACT	v
	LIST OF FIGURES	vii
	LIST OF TABLES	viii
1	INTRODUCTION	1
2	LITERATURE SURVEY	2
3	AIM AND SCOPE OF THE EXISTING DESIGN	
	3.1 EXISTING SYSTEM	6
	3.2 PROPOSED SYSTEM	7
	3.3 STANDARDS	8
	3.4 CONSTRAINTS	9
	3.5 TRADE OFFS	9
4	METHODOLOGY AND ALGORITHM USED	
	4.1 BLOCK DIAGRAM OF SELECTED METHODOLOGY	10
	4.2 ALGORITHM REQUIREMENTS AND DETAIL DESIGN APPLYING	12
	4.3 SOFTWARE – RANDOM FOREST ALGORITHM	16
5	RESULTS AND DISCUSSION	26
6	CONCLUSION	32
	REFERENCES	33

LIST OF FIGURES

FIGURE NO.	NAME OF THE FIGURE	PAGE NO.
3.1	Existing systems block diagram	6
3.2	Proposed system block diagram	7
4.1	Block diagram of the system	10
4.2	Random Forest algorithm flowchart	17
4.3	Overall design flowchart	22
4.4	Arduino UNO	23
4.5	DHT11 sensor	23
4.6	DS18B20 Temperature	24
4.7	Heart beat sensor	24
4.8	GSM modem	25
5.1	Project hardware model	26
5.2	Power supply for the project	27
5.3	Interfacing of the sensors with Arduino	27
5.4	LCD displaying temperature and Humidity level	28
5.5	Power supply to the GSM	28
5.6	Output in python shell	29
5.7	Output in thingspeak website	30
5.8	Sending alert to the user through GSM	31
5.9	Test data size accuracy test	31

LIST OF TABLES

TABLE No.	TABLE NAME	PAGE No.
2.1	Comparison of different healthcare assistance systems	5
5.1	Collected data in tabular form	30
5.2	Performance table	31

CHAPTER 1

INTRODUCTION

Embedded systems that are based on the Internet of things (IoT) are currently being transformed into networked smart gadgets that use detectors. The major disadvantage of utilizing connected devices was their small storage ability and faster processors, which had been alleviated by cloud services, which internationally recognized processing capabilities. To deal with smart health monitoring setups, it is necessary to integrate Internet of Things devices with cloud technology. Victims are now being watched in instantaneously in the intelligent medical system, according to the company. Supervised learning and machine intelligence are required for healthcare management and for achieving great and rapid treatment at a cheap cost. A smart framework is required for this (AI). AI, in conjunction with the Internet of Things-cloud architecture, has shown significant progress in offering a genuine intelligent medical system and increased judgment capabilities. Using detectors, modern mobile devices may assist in delivering a timely reaction from medical experts in a wise clinical setting. Individuals in emergency surgery need a prompt reaction, record data, and urgent treatment to remain alive and recover. It is impossible for medical specialists to remain on call around the clock in a hospital setting. As a result, to overcome these difficulties in healthcare systems, an intelligent framework is necessary.

Further investigation into this innovation will provide us with an idea of numerous wearable gadgets, such as a reusable patch, a bracelet, smart clothing, and so on. These instruments can measure heart rate, electrical pulse state, and diagnose various physical illnesses such as cardiovascular disease as well as other related illnesses. The collected data must be saved on the server to be used for process monitoring and control. This pproject studies the impacts of two technologies - IoT and AI – on healthcare through a systematic review of the literature. This is also meant to provide an overview of recent efforts in health monitoring and management.

CHAPTER 2

LITERATURE SURVEY

The field of e-health has increased remarkable progress in recent years, which is increasing by advancements in the Internet of Things and other technologies. From last 5 years, IoT has been using with combination of the Artificial Intelligence. We have seen a numerous projects on the health care that based on hardware and software technologies. Even though, IoT and AI based healthcare system is still in initial stages. It is lacking in terms of speed, connectivity, simplicity of the devices etc. In this literature survey, we researched about existed and existing healthcare systems which support the human health.

In 2022, Md Julhas Hossain et al. [1] discovered a project on Development of an IoT Based Health Monitoring for e-health. The authors developed a patient's body temperature and pulse rate monitoring system. In this they looked into IoT-based healthcare systems, which comprise sensors, smartphones, and microcontroller unit-based techniques, as well as the systems' working procedures, problems and constraints, and operating capability. The goal of the author is to offer a technological and economical model for an IoT-based medical system for easier and more comfortable medical services for all types of patients, as well as the impending problems and complications in adopting IoT in the real world's medical area. In project they used the temperature sensor (LM35), Pulse sensor, Arduino Uno, Wi-Fi module, ThingSpeak IoT cloud technology to process data and to computations on the collected data. Finally, they got graphs of the pulse rate.

In 2019, Tamanna Shaown et al. [2] They took the problem "heart attacks". They discovered a IoT based portable ECG monitoring for smart healthcare. The author developed a system a which can measure the heart rate and electricals signals of the heart. The collected data of electricals signals stored in the IoT cloud. Other users and doctors monitored the collected data from the cloud storage. This project provides the remote heart check-up of patient's who effected by the heart attacks. They used ECG sensor, Arduino Uno, Rasbery Pi. They used the ZigBee technology to connect the device to server. In this paper, Authors took the particular problem and provided a low-cost solution to reduce frequent health check-ups. But this system is very slow to process the data. So, it results difference in the accuracy of computed data and not focused on the surrounding environment conditions.

In 2019, Sonali Mahajan et al. [3] has designed a IOT based Smart Health Monitoring System for Chronic Diseases. This paper like above first paper. In this paper, they improved the use of the sensors. They focused on the problem to detect the falling of the old people. They used pulse oximeter (MAX30100), accelerometer sensor (ADXL345), temperature sensor (DS18B20). They used cost efficient board particle photon which is a medium range cost provide both storage and Wi-Fi facility. They used the third party MQTT server that is Adafruit to store sensor data in the cloud. It just feedbacks the values whatever the sensors read. It did not provide any predicting values of the health problems.

In 2021, Murali Krishna et al. [4] has designed a Smart Robotic System for Remote Health Monitoring of the Environment Using IoT and Global Positioning System. They proposed this to monitor the conditions of the environment (surroundings). They used different sensors like pressure, humidity, temperature, gas sensor. All these connected to the Arduino and Rasberry Pi which enables the Wi-Fi facility. This robot is helpful in emergency situations to monitor. But, the disadvantage of this project is it doesn't provide any AI based solutions. The collected data is useless.

In 2022, Vinayagam et al. [5] designed a project "An AI Tool for Emergency Medical Assistance System". This project attempts to check the medical condition of human beings by incorporating a smart medical assistance centre in public places such as pharmacies, shopping centres, transportation route stations, beside highways and areas where clinical stores are limited. The main advantage in their time is the people able to recognize their current health condition. It almost similar to chatbot but it gives the suggestions to the present health condition. In this project they used python IDE to develop a message feedback suggestion.

In 2021, Priyanka Dwivedi et al. [6] has designed a "IoT Enabled Smart Healthcare Assistance for Early Prediction of Health Abnormality". This paper is all about patient real time health monitoring with future predictions via mobile phones. They used the Machine Learning (ML), IoT, Cloud Computing (CC) for early prediction of health abnormality. It is the project combination IoT and AI. This project monitors patient's health but not the patients living room (surroundings). For a patient, environment around his/her are should be monitored to get prevent diseases.

In 2022, Skeeper named company designed an AIoT based stethoscope which is one of the biggest changes in the health sector. This stethoscope comes with AI and with storage, Wi-Fi accessible device. This device consists a display to see the graph. They mainly focusing on the heart and lung problems.

In 2020 HeroHealth named company designed a smart medication pill dispenser. It maintains the patient's medication carefully. Make contribution to be healthy. The main aim this project is to maintain correct medications and on time without confusion for patients. Disadvantage of this device is, It cannot be used by the multiple user only single user can be used and more power consumption.

V.Santhi et al. [7] proposed IoT based wearable device for monitoring health of pregnant ladies by reading environmental factors e.g. pressure, temperature and patient heartbeat. In case of abnormal factors their device will generate alarm and communicate with webApp using Wi-Fi. They used C4.5 decision trees classification algorithm for output prediction.

Sohini Roy Choudhury et al. [8] designed wearable device to monitor wound healing with help of uric acid biosensor. Their proposed device detects Uric Acid(UA) from wound. UA termed as biomarker, which have strong correlation with wounds and their healing. They used to be doxelectron shuttle, ferrocene carboxylic acid(FCA) which made transfer of electron between the enzyme and the transducer. For Uric acid detection in wound fluid they used wound fluid volume range as 0.5– 50 μ L. Their case Studies from different wound samples have shown an average recovery of 107%.

Hina Sattar et al. [9] proposed An IoT-based Intelligent Wound Monitoring System to monitor the wound condition and to enhance healing process. The proposed IoT based intelligent wound assessment system, for assessment of wound status and apply entropy and information gain statistics of decision tree to reflect status of wound assessment by categorization of assessment results in one of three class i.e. good, satisfactory or alarming. This system used IoT approach for measurement of real time wound factors i.e. Air temperature and humidity, body oxygen saturation and body temperature and predict effect of these factor on wound healing by doing classification with decision trees. Proposed system applied Entropy, information gain statistics to choose best split criteria for decision tree.

Table 2.1 Comparison of different healthcare assistance systems

S.No	Parameter	Control Board	Technology used	Bit rate	Range	AI and ML Tools
1.	Blood pressure, Heart beat	Arduino Uno	GSM, GPS, Thingspeak	270 kbps	Depends on cellular network connections	No tools used
2.	Temperature, Heartbeat	Arduino Uno, ESP8266 Wi-Fi module	Wi-Fi	2-4 mbps	30-200 meters	No tools used
3.	Heart beat(ECG)	Arduino Uno, Rasbery Pi	Wi-Fi/Zigbee/Bluetooth (GUI)	1.1-54 mbps, 10-250 kbps, 3-4 mbps	20-200 m, 2-20 m, 20-30m	No tools used
4.	Temperature, Pulse rate, Oxygen levels, Accelerometer (fall detection)	Particle Photon	Wi-Fi	5-10 mbps	200m	No tools used
5.	Heart rate, Blood glucose and cholesterol, ECG, BP, Temperature(body)	Wi-Fi based microcontroller	GSM	-	Depends on cellular network connections	ML and DL tools used
6.	Heart rate, Body temperature	Node MCU, Rasbery Pi	Wi-Fi	5-10 mbps	20-200 m	AI tools used
7.	Temperature, IR pulse	Arduino Uno, Wi-Fi module	GSM	250 kbps	Depends on cellular network	No tools used
8.	Heartbeat, Lungs(respiratory)	-	Wi-Fi	2-54 mbps	20-200 m	AI is used
9.	Body temperature, Air temperature, Humidity, Body oxygen saturation	Arduino Uno	-	-	-	-

CHAPTER 3

EXISTING SYSTEM AND PROPOSED SYSTEM

3.1 EXISTING SYSTEM:

In social insurance framework for patients who stays in home during post operational days checking is done either via overseer/ medical caretaker. Ceaseless observing may not be accomplished by this system, on the grounds that anything can change in wellbeing parameter inside of part of seconds and amid that time if guardian/attendant is not in the premises causes more noteworthy harm. So with this innovation created period where web administers the world gives a thought to add to another keen health awareness framework where time to time constant checking of the patient is accomplished.

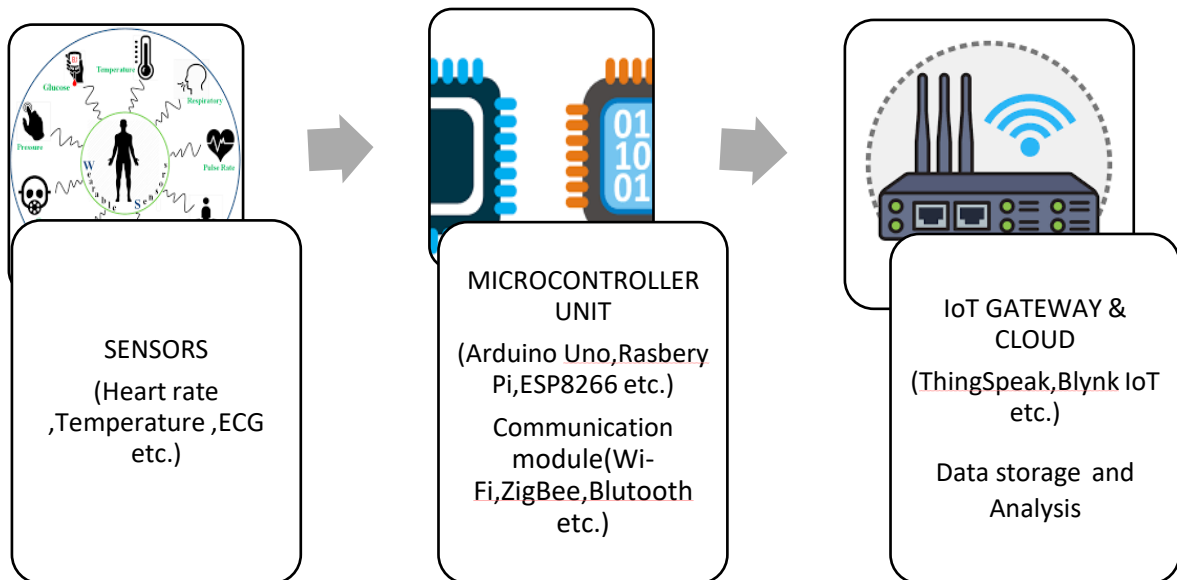


Fig:3.1 Existing systems block diagram

3.2 PROPOSED SYSTEM:

The proposed system is a comprehensive system that combines health monitoring with real-time environmental data. For example, integrate sensors for air quality, temperature, and humidity with health monitoring devices. Use AI to analyze how environmental factors may impact an individual's health over time. This could be particularly beneficial for individuals with respiratory conditions. By this system we will overcome problems in existing systems. The proposed will be the combination IoT and AI which is a new technology emerging now-a-days. In this project, we are going to improve the speed of the health monitoring systems with minimum latency. To facilitate data transmission and storage, with the power of cloud technology,

particularly Machine Learning tools, which offers a robust and secure platform for handling the extensive data generated by our system. The data collected from sensors is wirelessly transmitted to the cloud, where it is stored in dedicated databases. software components consist of AI algorithms responsible for data analysis and generating meaningful insights from the collected data. These algorithms employ machine learning techniques to recognize patterns, anomalies, and trends in the health metrics. As data is continuously streamed into the cloud, the AI system processes and analyzes it in real time, making it a valuable resource for healthcare providers. This system compares environment changes and Body environment changes. This will give awareness of the human health condition. So that can prevent some basic diseases like respiratory problems, Fever, Cold etc. As data is continuously streamed into the cloud, the AI system processes and analyses it in real time, making it a valuable resource for healthcare providers.

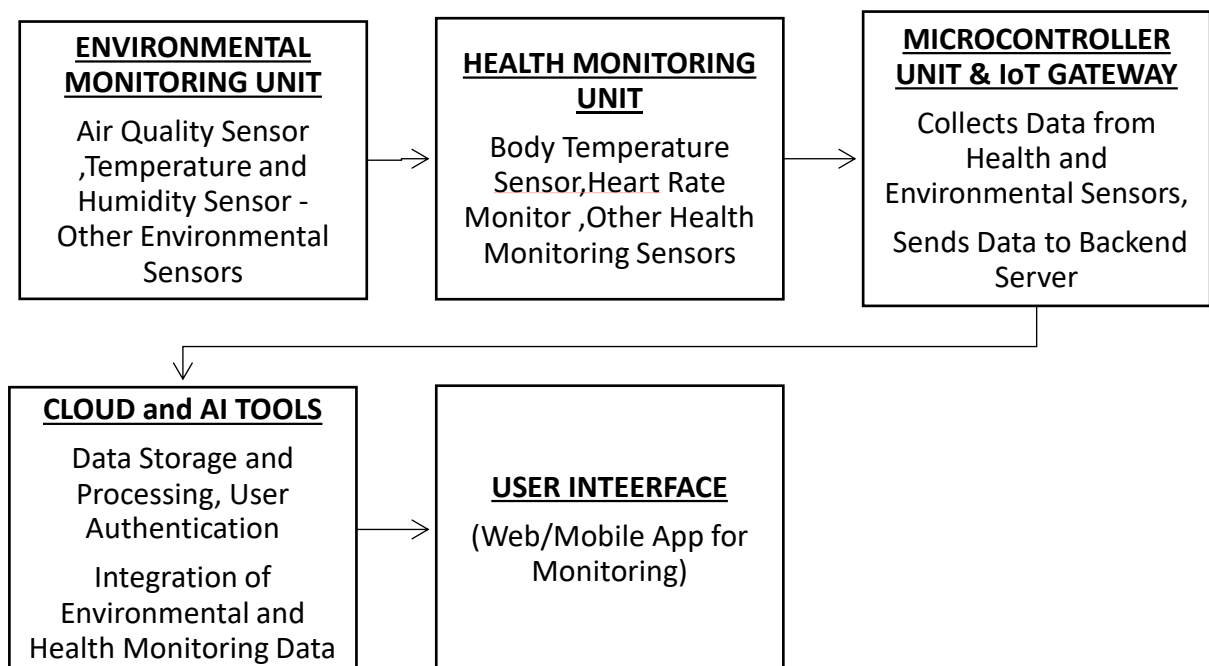


Fig:3.2 Proposed system block diagram

STANDARDS:

Ensured that all the hardware components (Arduino Uno, sensors, GSM module) are compatible in terms of electrical requirements, such as voltage levels and signal interfaces. This prevents damage to components and ensures proper communication between devices. All the components used in this project are manufactured from certified companies which follows electronic standards

All the sensors used in this project are noise compatible and can be compatible with Arduino Uno. Chosen hardware components that prioritize data security and encryption to protect sensitive health information transmitted between devices and servers. Ensured that the chosen GSM module supports secure communication protocols, such as HTTPS protocol used in this project. This GSM will send message to only particular registered user. Selected energy-efficient hardware components use minimized power consumption, especially for devices powered by regulated power supply. Used sensors are low-power sensors and optimized the power usage of the microcontroller (Arduino Uno) to conserve energy. Considered the operating environment where the devices will be deployed. Chosen hardware components that are durable and capable of withstanding temperature variations, humidity, and other environmental factors commonly found in healthcare settings. The designed hardware architecture provides scalability to accommodate future expansion or upgrades. Guarantee that the selected hardware components support expansion interfaces, such as GPIO pins or serial communication ports, for connecting additional sensors or peripherals.

CONSTRAINTS:

Hardware Limitations: The Arduino Uno has limited processing power and memory, which may constrain the complexity of data processing and analysis that can be performed onboard. IoT devices are often battery-powered, so power consumption must be carefully managed to prolong battery life. The GSM module may have limitations on data transmission rates and coverage, affecting the speed and reliability of alert messages sent to healthcare providers. The accuracy and precision of sensor readings (temperature, heartbeat, humidity) may vary, introducing uncertainty into the data analysis process. All sensor are works normally in the rated temperature ranges. Accuracy cannot be increased more than the rated accuracy. This project is critical to use in rainy weather conditions. Project deployed

location (environment) should provide better network for GSM module to send alert messages. The compactness of the project is directly proportional to the expenditure spent on the project. Project should be placed in static condition in a location to work effectively and accurately.

TRADE OFFS:

Higher-performing sensors or more advanced hardware components may improve system performance but could also increase project costs. Increasing the complexity of the machine learning model (e.g., using more sophisticated algorithms or additional features) may improve prediction accuracy. Real-time data collection provides immediate monitoring over previously stored data. Algorithm used in this project provide better computations on the collected data from all the sensors. Collecting more detailed patient data (e.g., continuous monitoring of surrounding temperature, heartbeat, Body temperature) may enhance the accuracy of health predictions but raises privacy concerns and may require explicit user consent.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 Block diagram of Selected Methodology:

Main purpose of IOT based system to help healthcare system in case of emergencies. The system self-monitor and able to inform critical situations of patients to the doctors. Sensors signals are sent to Arduino. Here patients body temperature, heartbeat values is measured using respective sensors and it can be monitored in the cloud database system as well as monitored anywhere in the world using internet source. If any sensor value crossed its limit, then an SMS will be sent to the concerned person's. Monitoring values will be checking to prediction by machine learning.

Block diagram:

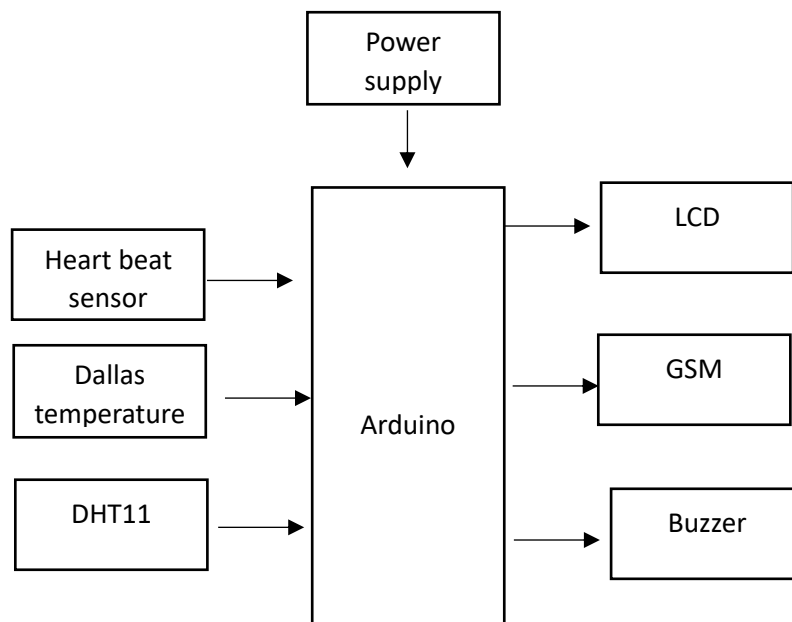


Fig:4.1 block diagram

LIST OF COMPONENTS REPRESENTED IN BLOCK DIAGRAM

- Arduino
- DHT 11 (temperature and humidity) sensor
- Dallas temperature sensor
- Heart beat sensor
- 16x2 LCD

- GSM
- Buzzer

The healthcare assistance system provided in the block diagram is a complete solution designed to monitor vital health parameters and environmental conditions for individual's requirement of medical attention or surveillance. This system integrates various components, including sensors, a microcontroller, communication modules, and user interface elements, to provide real-time monitoring, data processing, and alerting capabilities. Function of each block given below.

1.Power supply: A stable and reliable power supply is essential for the continuous operation of electronic systems, particularly which are used for critical applications such as healthcare monitoring. The power supply unit ensures that all components receive the necessary voltage and current levels for proper functioning. In this healthcare assistance system we integrated the power supply with GSM module for all the components regulated power adapter serve as the power source, supplying power to the central processing unit Arduino microcontroller.

2.Heart beat sensor: A heartbeat sensor also known as a pulse sensor, is the main component in healthcare assistance systems for monitoring heart rate or heart rate variable readings. Connecting this sensor to an Arduino microcontroller enables real-time monitoring and analysis of heart-related data in a real time. We connect this to the finger of the user to detect the heartbeat.

3.Dallas temperature sensor: The Dallas Temperature Sensor plays a crucial role in monitoring body temperature. Body temperature is a important parameter that can indicate various health conditions including fever and headache. Monitoring changes in body temperature in real-time enables early detection of abnormal conditions of the humans.

4.DHT11: The DHT11 is a temperature and humidity sensor capable of providing accurate readings in various environmental conditions. It consists of a humidity sensing element and a thermistor for temperature measurement. The sensor communicates with the Arduino microcontroller using a single-wire digital protocol.

5.LCD: It is the user interface which plays a crucial role in providing feedback in real-time. In the healthcare assistance system, a Liquid Crystal Display (LCD)

serves as the primary user interface, displaying the important information such as sensor readings, system status and alerts.

6.GSM: The GSM (Global System for Mobile Communication) modem which is used to send the alerts in the form of text message to the phone number provided by the user.

7.Arduino: It is the main heart of the project because all these components are connected to the microcontroller Arduino uno R3. It serves as the central processing unit responsible for data collection, processing, decision-making, and control. Equipped with analog and digital input/output pins, the Arduino interacts with various sensors to collect data, processes the required information, and executes predefined algorithms logic to determine the user's health status. Additionally, the Arduino makes communication with external devices such as the GSM module for alerting.

4.2 Algorithm requirements and detail design applying:

The circuit comprises three sensors interfaced with an Arduino Uno microcontroller.

DHT11 Sensor: The DHT11 sensor measures temperature and humidity. It sends digital signals to the Arduino Uno, which processes these signals to obtain temperature and humidity readings.

Clip Type Heartbeat Sensor: This sensor measures heart rate by detecting changes in blood volume in a finger. It provides analog signals to the Arduino Uno, which then interprets these signals to calculate heart rate.

Dallas Temperature Sensor: The Dallas temperature sensor provides digital temperature readings to the Arduino Uno. It uses the OneWire protocol to communicate with the Arduino Uno.

Arduino Algorithm:

The Arduino algorithm for this circuit involves the following steps:

Initialization: Initialize the Arduino Uno and configure the pins for sensor input and output.

Sensor Reading: Read data from each sensor connected to the Arduino Uno. This includes reading temperature and humidity from the DHT11 sensor, heart rate from the clip-type heartbeat sensor, and temperature from the Dallas temperature sensor.

Data Processing: Process the sensor data as required. This may involve converting analog readings to meaningful units (e.g., temperature in Celsius), filtering noise, or performing calculations (e.g., heart rate calculation).

Output: Display or transmit the processed sensor data. This could be done through serial communication for real-time monitoring, displaying on an LCD screen, or transmitting wirelessly to another device.

Repeat: Continuously loop through the steps above to continuously monitor the sensors and provide real-time data.

This algorithm allows the Arduino Uno to effectively interface with the sensors, collect data, and perform desired actions based on the sensor readings.

Data Processing:

After obtaining readings from the sensors, the Arduino Uno processes this data to make it usable and interpretable. This involves several steps depending on the nature of the sensor and the desired output. For instance:

Temperature and Humidity Data from DHT11 Sensor: The raw data received from the DHT11 sensor needs to be converted into understandable temperature and humidity values. The Arduino Uno applies mathematical formulas or algorithms to convert the digital signals into Celsius (for temperature) and percentage (for humidity). Additionally, error-checking mechanisms might be implemented to ensure the accuracy of the readings. The analog signal output by the heartbeat sensor corresponds to the variations in blood volume, which are indicative of the heart rate. The Arduino Uno algorithm processes these analog signals to calculate the heart rate in beats per minute (BPM). This process might involve signal filtering to remove noise and improve accuracy. The Dallas temperature sensor provides digital temperature readings using the OneWire protocol. The Arduino Uno algorithm interprets these readings and possibly performs additional processing, such as converting the readings into Celsius or Fahrenheit and calibrating for accuracy.

Once the sensor data has been processed, the Arduino Uno can generate output in various forms to make the information accessible to users or other systems.

Common methods of output include:

Serial Communication: The Arduino Uno can transmit the processed sensor data via serial communication to a connected computer or another microcontroller. This allows real-time monitoring of sensor readings and enables further analysis or control actions.

LCD Display: An LCD (Liquid Crystal Display) screen connected to the Arduino Uno can be used to visually display the sensor readings. The Arduino Uno sends formatted data to the LCD, which then displays it in a human-readable format. This provides a convenient way to view the data without the need for additional devices.

Wireless Transmission: Using wireless communication modules such as Bluetooth or Wi-Fi, the Arduino Uno can transmit sensor data to remote devices or a central monitoring system. This enables monitoring and control of the sensors from a distance, making it suitable for applications where wired connections are impractical.

The Arduino Uno continuously loops through the steps of sensor reading, data processing, and output generation to provide ongoing monitoring and control of the connected sensors. This iterative process ensures that the system remains responsive to changes in sensor readings and can adapt its behaviour accordingly.

Libraries and Definitions:

- The code begins by including necessary libraries like OneWire, DallasTemperature, PulseSensorPlayground, LiquidCrystal, and DHT. These libraries provide pre-written functions to interact with sensors and devices connected to the Arduino.
- Definitions are made for various pins and parameters used in the code. For instance, `ONE_WIRE_BUS` defines the pin connected to the Dallas temperature sensor, `PulseWire` defines the pin connected to the pulse sensor, and `DHTPIN` defines the pin connected to the DHT11 sensor.

Set Up Function:

In the `setup()` function, the Arduino initializes necessary components and sensors. It starts communication with the serial monitor (`'Serial.begin(9600)'`), the GSM module (`'gsm.begin(9600)'`), DHT sensor (`'dht.begin()'`), pulse sensor (`'pulseSensor.begin()'`), and LCD (`'lcd.begin(16, 2)'`). The threshold for the pulse sensor is set (`'pulseSensor.setThreshold(Threshold)'`). The LCD displays a welcome message for 2 seconds.

Loop Function:

- The `'loop()'` function continuously runs the main program.
- It checks if there is any data available from the serial monitor (if (`'Serial.available() > 0'`)), and if so, it reads a character and executes a corresponding command (`'executeCommand(command)'`).

- It reads temperature and humidity from the DHT sensor and converts temperature from Celsius to Fahrenheit.
- It requests temperature readings from the Dallas temperature sensor ('sensors.requestTemperatures(')).
- It displays sensor readings on the LCD, including body temperature, heartbeat, room humidity, and room temperature.
- It constructs a string containing sensor readings and sends it via the serial monitor.
- Depending on the received command, it may send SMS messages indicating abnormal heart rate or temperature detected.

ExecuteCommand Function:

- The 'executeCommand()' function is called when a command is received from the serial monitor.
- Depending on the command received ('1' or '2'), it sends an SMS message via the GSM module to a predefined phone number, indicating either abnormal heartbeat or abnormal temperature detected.
- It also triggers the buzzer briefly to indicate message sending.

We need to send sensor data to ThingSpeak's servers for visualization and analysis. We will use the ThingSpeak library to accomplish. The data is formatted as HTTP queries with appropriate field values and sent to the ThingSpeak server using the `requests.get()` function.

The code utilizes software serial communication ('SoftwareSerial') to communicate with the GSM module. The PulseSensorPlayground library is used for interfacing with the pulse sensor. The LCD is used to display information about the system's operation and status. Overall, the algorithm sets up a system that monitors body temperature, heartbeat, room humidity, and room temperature using various sensors connected to an Arduino Uno. It sends SMS alerts if abnormal readings are detected. For example, if an abnormal heart rate is detected, a command is sent via serial communication to the connected device (e.g., Arduino) to trigger an alert (e.g., sound a buzzer). Similarly, if abnormal temperature is detected, a command is sent to trigger a different action.

HARDWARE: The working of hardware fully based on the sensors used in this project and Arduino uno. For body temperature, temperature is measured in

Fahrenheit by converting the sensed data from Celsius. $F = C \times 9/5 + 32$. Surrounding temperature is measured in Celsius. All the required libraries and functioning process code of sensors are uploaded into the Arduino uno. API key of the ThingSpeak is also written in the code to show graphical representation. Total hardware is to sense data from surroundings. The GSM module is the communicator between system and user. When system crosses the normal range values, GSM module sends the 'Take care!' message.

4.3 SOFTWARE

Data is collected from hardware by using the USB cable. In python, serial port is connected by using pyserial3.5 library. So, it can directly get data from sensors. Openpyxl library used to form csv file to do computations with collected data. Random forest algorithm is written in the python. Initially we need to separate training data and test data using train test split library. Random forest classifier is used to do predict the output. Working principle of random forest algorithm is ensemble learning. Without knowing what the data was given, by collection (voting) of different decisions of trees it finally take one decision according to majority. Sklearn consists the ensemble technique and this is imported to the random forest classifier. This algorithm provides flexibility in computations. It can handle large data compare seasonal Arima [6]. it takes long time to learn, mainly used for seasonal pattern changes. RFA is fast learning algorithm. It quickly learns from data and give accurate output.

In machine learning, this concept of multiple models working together to come to an aggregate prediction is called ensemble learning. It provides the basis for many important machine learning models, including random forests.

Ensemble learning creates a stronger model by aggregating the predictions of multiple weak models, such as decision trees (check out our previous article on Decision Trees to learn more about some of their limitations). Condorcet's Jury Theorem suggests that the majority vote aggregation can have better accuracy than the individual models. There are other methods to aggregate predictions, such as weighted majority vote. Random Forest is an example of ensemble learning where each model is a decision tree. In the next section, we will build a random forest model to classify if a healthcare system is detecting the abnormal situations. These abnormal situations come in many variations, and we will use four simple features:

heartbeat, humidity, surrounding temperature, Body temperature, and if sensed data has crossed normal range.

One way to produce multiple models that are different is to train each model using a different training set. The Bagging (Bootstrap Aggregating) method randomly draws a fixed number of samples from the training set with replacement. This means that a data point can be drawn more than once. This results in several training sets that are different.

Here are the features: Humidity, Body temperature, surrounding temperature, Heartbeat and take care or not. For each split of the tree building, we compute the best splitting using only a randomly selected subset of the features. This is another way to ensure that the decision trees are as different as possible.

Let's build the first tree with the first sample! For the first split, 3 features (hum, bt, st) are selected to find the best split.

The best feature to split is body temperature. Likewise, for the second split, another set of three features (hum, st, hb) are selected to find the best split. The best feature to split this time is "heartbeat." This process continues until the tree is constructed. The other samples will build other trees using the same process. There many trees built from different samples. Based on this training data algorithm is learnt. To test the trained algorithm test data can be used. By this each tree take a decision.

Different trees can give different decisions according to the training data. To take

Steps involved in Random forest algorithm

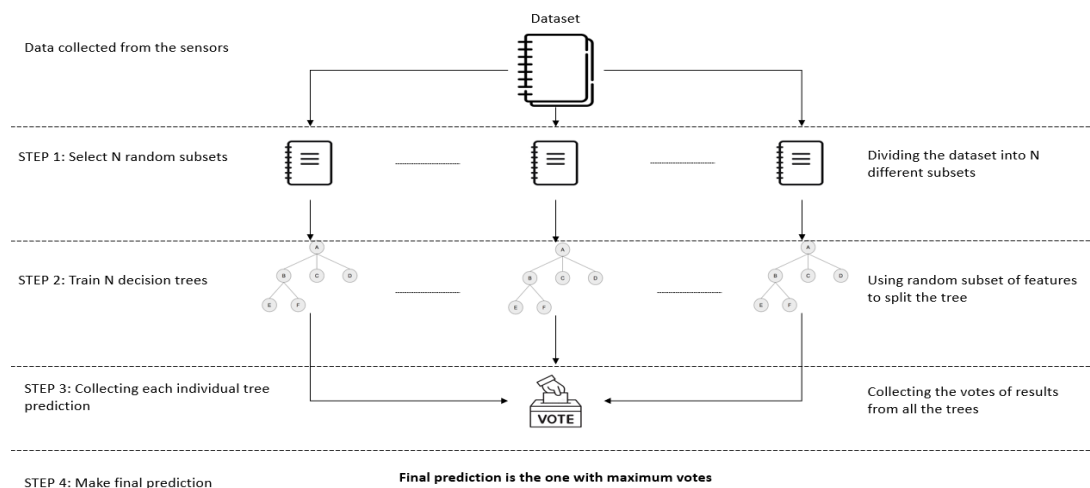


Fig:4.2 Random Forest algorithm flowchart

final decision counting should be taken from all the trees in the forest. According to the count which output feature has high count value it will predict as output. In

decision tree model suffers from high variance. However, this variance among trees is employed in the random forest as a feature, not a bug. The inventor of the random forest model Leo Breiman says in his paper "your results indicate that better (lower generalization error) random forests have lower correlation between classifiers and higher strength." The high variance of the decision tree model can help keep the correlation among trees low. The Bagging Method as well as the Feature Selection are the key innovations to keep correlation low.

The process of building and training the random forest classifier was carried out using Python-based Scikit-learn open-source library. Additionally, NumPy and Pandas libraries were imported for use in numeric operations and dataset management, respectively.

Functions used in python code

numpy: 'numpy' is a library used for numerical computations in Python. It provides support for large, multi-dimensional arrays and matrices, along with a collection of mathematical functions to operate on these arrays. Commonly used functions include array creation, manipulation, linear algebra, Fourier transforms, and random number generation.

pandas: 'pandas' is a library used for data manipulation and analysis in Python. It offers data structures and functions for working with structured data, such as tables or relational databases. Key components include the DataFrame (for tabular data) and Series (for one-dimensional labelled data), along with methods for data cleaning, reshaping, merging, and analysis.

Sklearn: sklearn (short for scikit-learn) is a library used for machine learning tasks in Python. It provides a wide range of supervised and unsupervised learning algorithms, along with tools for model evaluation and selection, data preprocessing, and feature engineering. Commonly used modules include model_selection for splitting data into training and testing sets, ensemble for implementing ensemble methods like Random Forest, and various other modules for classification, regression, clustering, and dimensionality reduction.

RandomForestClassifier(): RandomForestClassifier is a machine learning algorithm used for classification tasks. It's part of the ensemble learning methods, specifically the Random Forest algorithm. Random Forest builds multiple decision trees during training and merges their predictions to improve accuracy and reduce overfitting. Each decision tree in the forest is trained on a random subset of the training data

and selects a random subset of features at each split. RandomForestClassifier is used to build models for predicting whether sensor readings indicate normal or abnormal conditions. The code trains separate RandomForestClassifier models for each sensor parameter: room temperature, room humidity, body temperature, and heart rate. Each model is trained using labelled data from an Excel sheet, where the labels indicate whether the sensor readings are normal or abnormal. For example, the model for room temperature ('model_1') is trained using the dht_temp readings as features and the corresponding label_temp as the target variable. Similarly, models for room humidity, body temperature, and heart rate ('model_2', 'model_3', 'model_4') are trained using their respective features and target variables. Once trained, these models can be used to predict whether new sensor readings are normal or abnormal.

Making Predictions with RandomForestClassifier:

After training, the RandomForestClassifier models are used to make predictions on new sensor readings obtained from the serial port. For example, the code uses model_1.predict() to predict whether the current room temperature ('dht_temp') indicates a normal or abnormal condition. Similar predictions are made for room humidity, body temperature, and heart rate using their respective models ('model_2', 'model_3', 'model_4'). The predictions are then used to take appropriate actions based on the detected abnormal conditions, such as triggering alerts or sending commands to external devices.

```
X_train_1, X_test_1, y_train_1, y_test_1 = train_test_split(feature_1, label_1,
test_size=0.2, random_state=42)
```

```
X_train_2, X_test_2, y_train_2, y_test_2 = train_test_split(feature_2, label_2,
test_size=0.2, random_state=42)
```

```
X_train_3, X_test_3, y_train_3, y_test_3 = train_test_split(feature_3, label_3,
test_size=0.2, random_state=42)
```

```
X_train_4, X_test_4, y_train_4, y_test_4 = train_test_split(feature_4, label_4,
test_size=0.2, random_state=42)
```

Above statements tells about the training data and testing data. Feature_1,2,3,4 are input features, label are targeted output for corresponding feature. These statements are defined in the 'train_test_split' function from sklearn library. Test size 0.2 represents that 20% of data is considered for testing and 80% of dataset is considered as training data. Random state = 42 ensures that the data is split in the

same way each time the code is run, making the results reproducible. This is similar for all 4 features.

```
1.model_1 = RandomForestClassifier(random_state=42)
2.model_1.fit(X_train_1.values.reshape(-1, 1), y_train_1)
3.model_2 = RandomForestClassifier(random_state=42)
4.model_2.fit(X_train_2.values.reshape(-1, 1), y_train_2)
5.model_3 = RandomForestClassifier(random_state=42)
6.model_3.fit(X_train_3.values.reshape(-1, 1), y_train_3)
7.model_4 = RandomForestClassifier(random_state=42)
8.model_4.fit(X_train_4.values.reshape(-1, 1), y_train_4)
```

Here random forest classifier is a function defined in sklearn library. Line-1 initializes the random forest classifier object named model_1. Line-2 is which trains model_1 using the training data X_train_1, respective labels y_train_1. This is done for one feature. It is similar for remaining features.

Serial: The serial library is used for serial communication in Python. It provides tools for reading and writing data to and from serial ports, allowing communication with external devices such as microcontrollers (e.g., Arduino), sensors, and other hardware. Functions like Serial are used to create serial port objects, readline() reads data from the serial port, and write() writes data to the serial port.

Time: The time module provides various functions for working with time-related operations in Python. Functions like sleep() are used to introduce delays in the program execution, time() returns the current time in seconds since the epoch, and strftime() converts a time tuple to a string representing the time.

re: The re module is used for working with regular expressions in Python. Regular expressions are patterns used to match character combinations in strings, allowing for powerful text processing and manipulation. Functions like findall() and search() are used to search for matches of a pattern within a string.

Warnings: The warnings module provides functions for issuing warning messages in Python. Warnings are used to alert users about potential issues or problematic situations in their code. Functions like filterwarnings() can be used to control how warnings are displayed or handled.

Requests: The requests library is used for making HTTP requests in Python. It provides a high-level interface for sending HTTP requests and receiving responses

from web servers. Functions like `get()` are used to send GET requests, `post()` for POST requests, and `status_code` to check the HTTP status code of the response.

Loading excel sheet in python:

The Excel sheet is loaded into Python using the `read_excel()` function from the pandas library.

```
data = pd.read_excel("data.xlsx", engine="openpyxl")
```

`pd.read_excel()` is a function provided by the pandas library to read data from Excel files. The first argument "data.xlsx" specifies the path to the Excel file to be loaded. The `engine="openpyxl"` argument specifies the Excel engine to use for reading the file. `openpyxl` is one of the supported engines in pandas.

The function `pd.read_excel()` returns a `DataFrame`, which is a two-dimensional labelled data structure with columns of potentially different types. The `DataFrame` 'data' now contains the data from the Excel sheet, with rows and columns corresponding to the rows and columns in the Excel file. Each column in the Excel sheet becomes a column in the `DataFrame`, and each row becomes a row in the `DataFrame`. Once the Excel sheet is loaded into the `DataFrame` data, you can access and manipulate the data using various `DataFrame` methods and attributes provided by pandas. For example, you can access specific columns or rows, perform calculations, filter data.

The below flow chart represents the working of the total system. When the power supply is connected to the hardware kit the regulated power supply board distributes the power to the all components in the hardware and the sensors, Arduino board,gsm module is turned on. The code is already uploaded in Arduino then it starts collecting data from sensors. The data was also sent to the python ide using serial communication. According to the random forest algorithm which is written in python decision will be taken. If the sensor readings are abnormal then GSM gets activated. It sends an alert message to the user.Also the data is continuously uploaded into the cloud.Here we used the Thingspeak cloud platform.In python we used the request module for sending the data to the thingspeak cloud and also we can visualize the data in pictorial form such as graphs.If one can want the data in the form of tabular form then he can directly export the data from the website by clicking on export as cv.

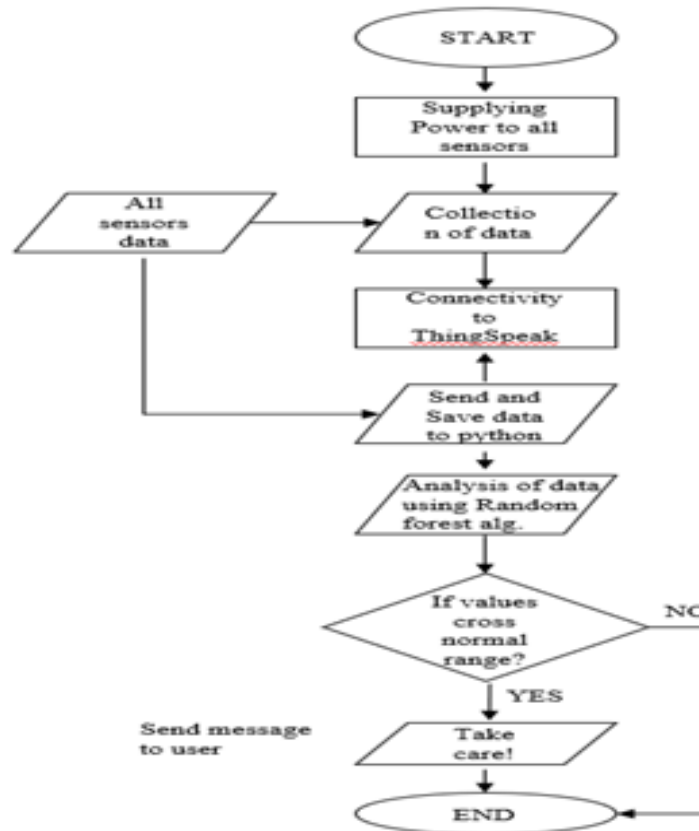


Fig:4.3 Overall design flowchart

HARDWARE IMPLEMENTATION OF PROPOSED SYSTEM

The hardware components used in the project are listed below-

1. Arduino
2. DHT 11 (temperature and humidity) sensor
3. Dallas temperature sensor
4. Heart beat sensor
5. 16x2 LCD
6. GSM
7. Buzzer

1. Arduino: In this project we used Arduino uno R3. The Arduino microcontroller is an easy to use yet powerful single board computer that has gained considerable traction in the hobby and professional market. The Arduino is open-source, which means hardware is reasonably priced and development software is free. The Arduino Uno R3 is a microcontroller board based on the ATmega328P chip, commonly used for prototyping and electronics projects. It features digital and

analog input/output pins, USB connectivity for programming and power, and is compatible with a variety of sensors and actuators. Its simplicity, versatility, and large community support make it a popular choice for beginners and experienced peoples.

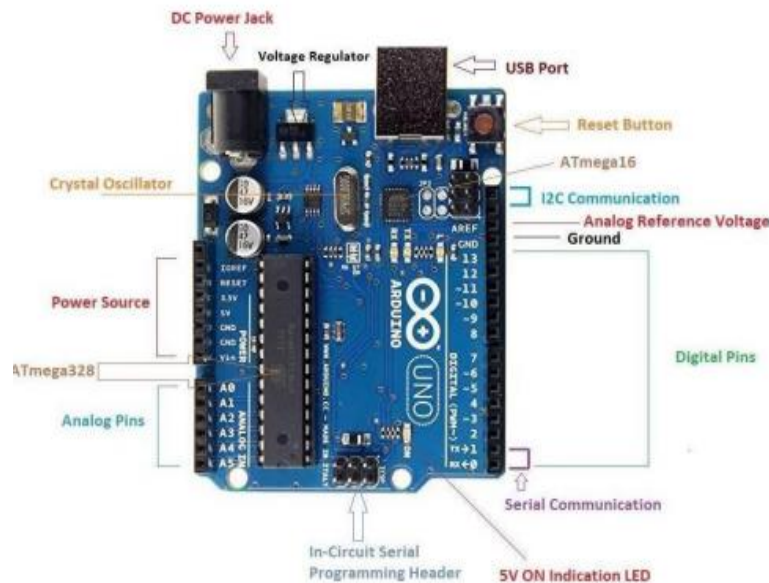


Fig:4.4 Arduino UNO

2. DHT11(Temperature and Humidity) sensor: The DHT11 is a basic, low-cost digital temperature and humidity sensor. It uses a capacitive humidity sensor and a thermistor to measure the surrounding air, and spits out a digital signal on the data pin (no analog input pins needed). It's fairly simple to use, but requires careful timing to grab data. The only real downside of this sensor is you can only get new data from it once every 2 seconds.

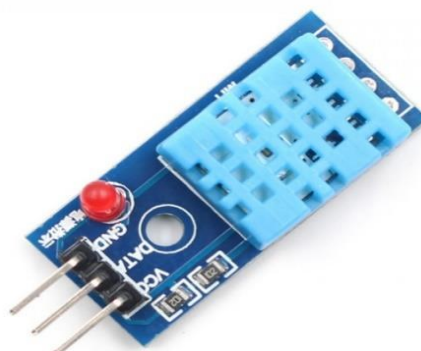


Fig:4.5 DHT11 sensor

3. Dallas Body temperature sensor: The digital temperature sensor DS18B20 follows single wire protocol and it can be used to measure temperature in the range of -67°F to +257°F or -55°C to +125°C with +5% accuracy. The range of received data from the 1-wire can range from 9-bit to 12-bit. Because, this sensor follows the single wire protocol, and the controlling of this can be done through an only pin of Microcontroller.



Fig:4.6 DS18B20 Temperature sensor

4. Heart beat sensor: Heartbeat Sensor is an electronic device that is used to measure the heart rate i.e. speed of the heartbeat. Monitoring body temperature, heart rate and blood pressure are the basic things that we do in order to keep us healthy. In order to measure the body temperature, we use thermometers and a sphygmomanometer to monitor the Arterial Pressure or Blood Pressure. The heartbeat is measured in beats per minute or bpm, which indicates the number of times the heart is contracting or expanding in a minute.

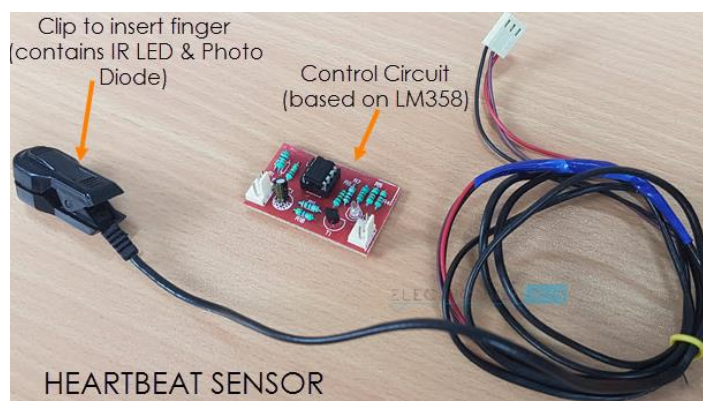


Fig:4.7 Heartbeat sensor

5. GSM (Global System for Mobile Communication): GSM (Global System for Mobile communications) is an open, digital cellular technology used for transmitting mobile voice and data services. It is a digital mobile telephone system that is widely used in Europe and other parts of the world. Here it is used to send the text messages to alert the abnormal conditions of the person.



Fig:4.8 GSM modem

CHAPTER 5

RESULTS AND DISCUSSION

The project simulation results and the hardware kit of the “smart health care assistance system ” are shown in this chapter by bringing it's goal.

5.1 Project kit

After interfacing all the modules as per the proposed system and connected to the power supply the fig 5.1 shows the hardware kit of the project.

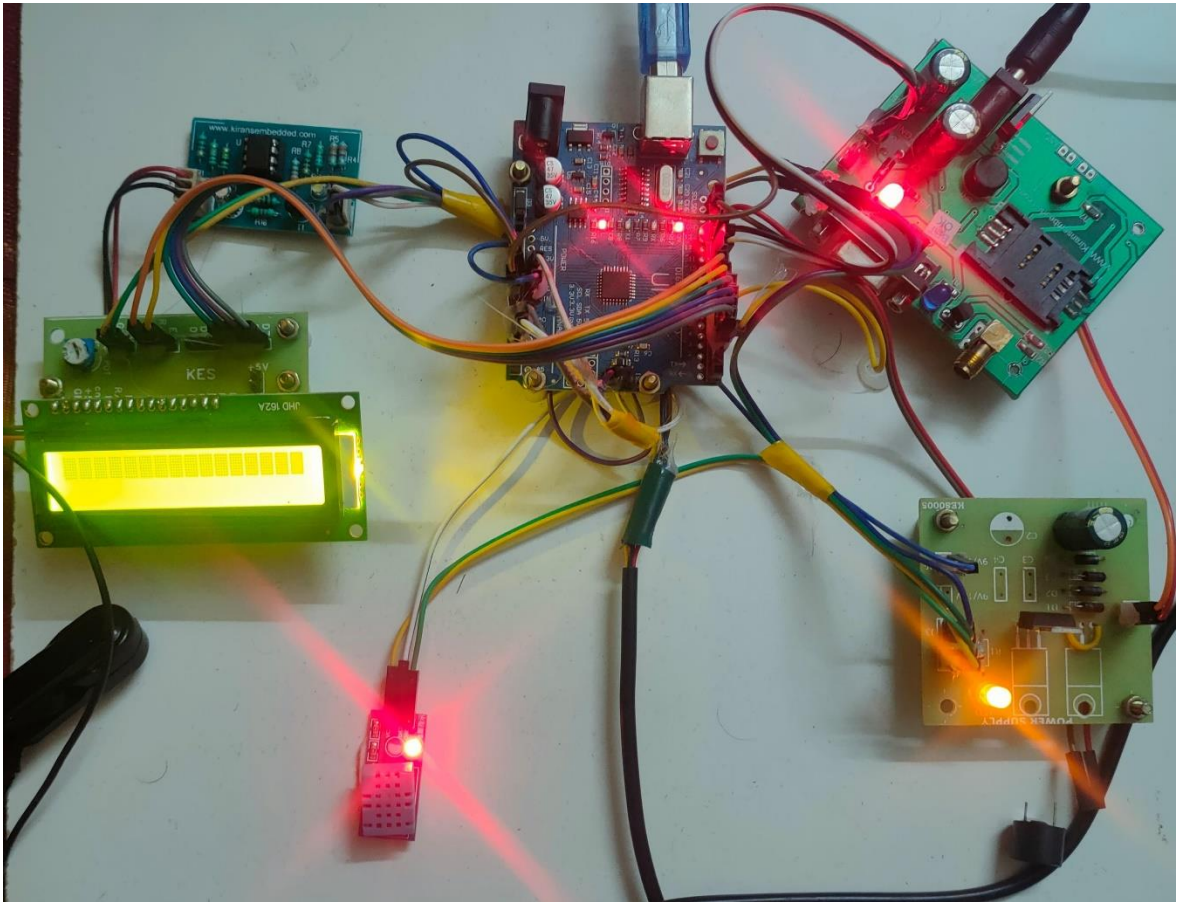


Fig:5.1 Project Hardware model

In the above figure it shows that how the connections are made and how they are interfaced with the Arduino uno microcontroller board was connected to GSM modem, Power supply board, 16x2 LCD Screen, Heart beat sensor, Body temperature, DHT11 temperature and humidity sensor.

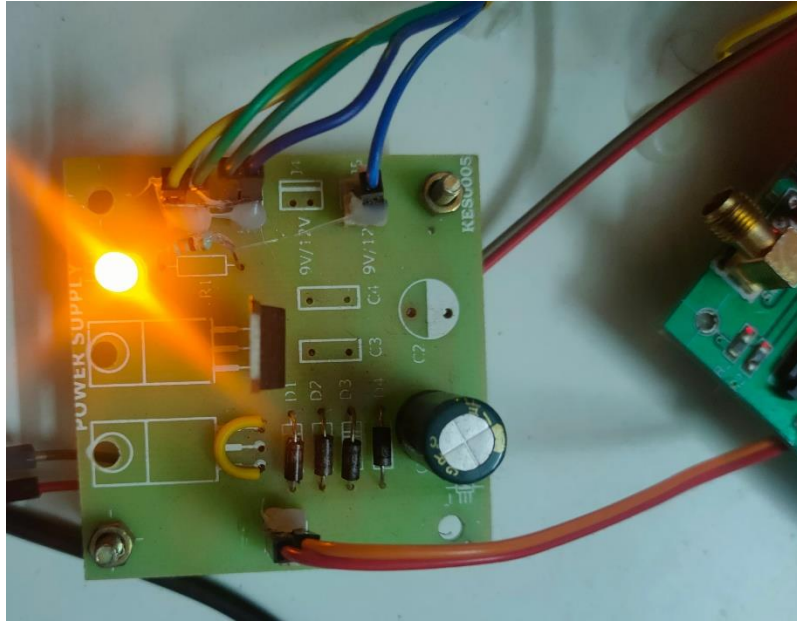


Fig:5.2 Power supply for the project.

The above fig 5.2. Shows the power supply board of the hardware kit. It is used to provide the power supply to the all sensors since Arduino boards typically have a 5V pin and a 3.3V pin that can be used to supply power to sensors and other peripherals. But we have used GSM module and LCD we need more power supply to the circuit. So it is used to distribute the power supply to all components which is taken from the GSM module power.

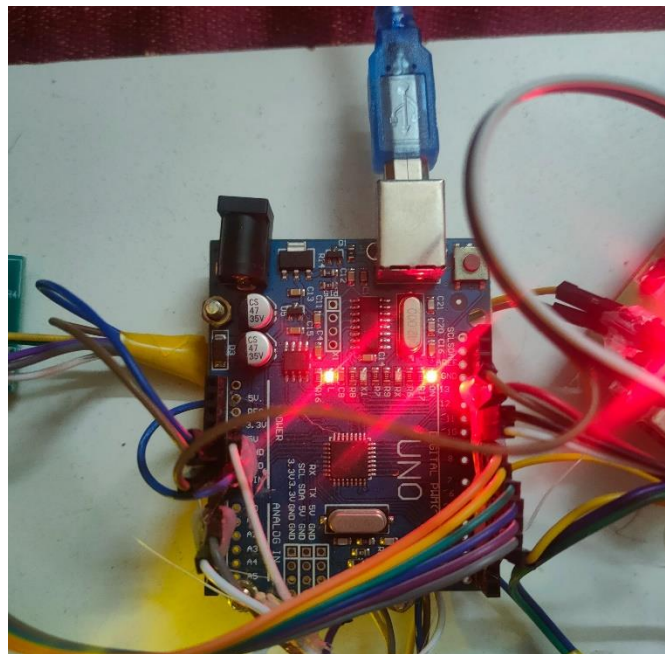


Fig:5.3 Interfacing the sensors with Arduino.

After interfacing all the sensors with Arduino and switching on the power supply the sensors starts collecting the data from the user and sends it to the Arduino in the form of the analog signal. These sensors are connected to the analog pins of the Arduino board so that the Arduino can read the signal sent by the sensors which are analog in nature. After reading the data from the sensor the Arduino microcontroller process the data and it is displayed on the LCD screen.

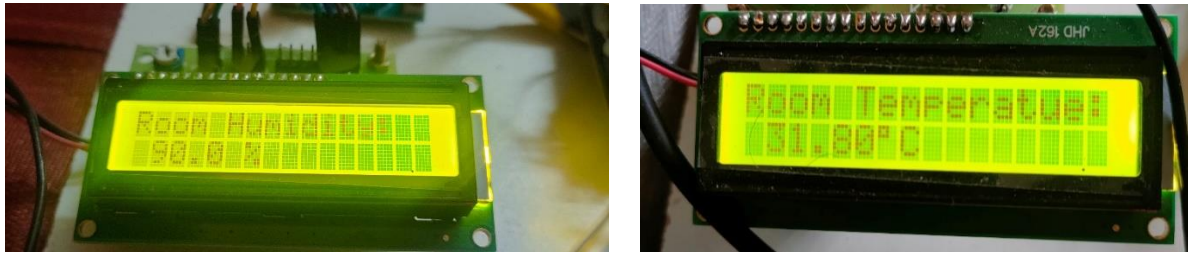


Fig:5.4 LCD displaying room temperature and Humidity level.

In this project there is also GSM modem. It is in the idle state until the parameters showing the abnormal conditions. Once there is any abnormal change in the readings such as the Body temperature or heart beat it will send the alerts to the provide mobile phone in the form of the SMS. In order to use this feature the user should install the sim card in the modem.

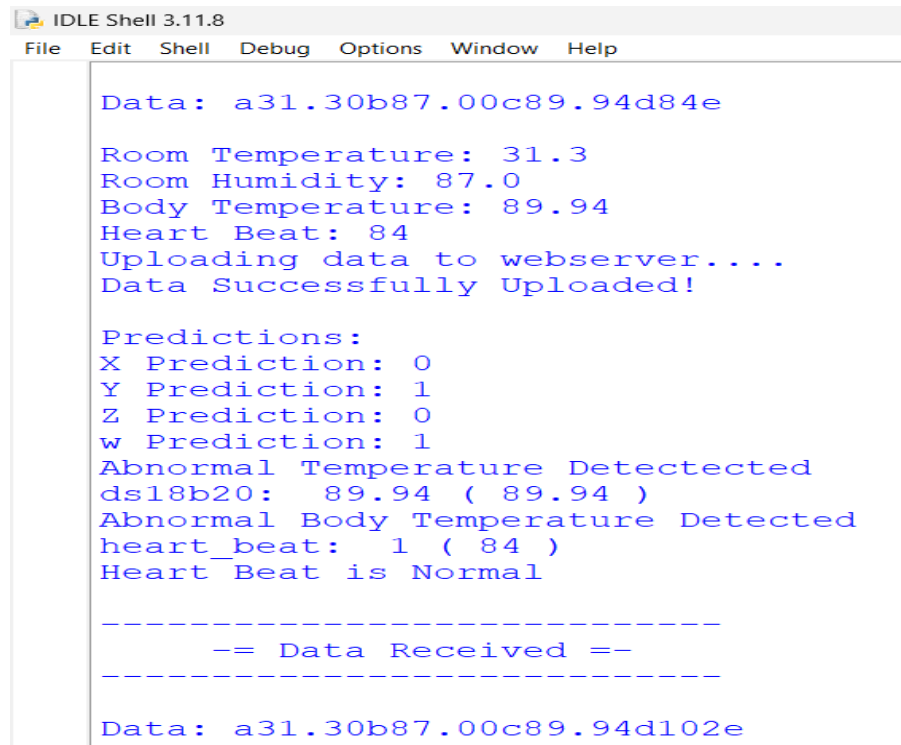
NOTE: Since GSM supports the second generation 2G and third generation 3G only so except JIO SIM the user can use any other SIM.

Also the power supply that is provided from the Arduino is not sufficient for the GSM modem. So we have to use an external adapter to provide the power supply to the GSM modem for its efficient usage.



Fig:5.5 Power supply to the GSM modem.

After providing all the power supply to the hardware kit then connect the arduino board to the laptop through the data cable. After the connection you can see the result in the python shell after running the python code. The fig 5.6 shows the results that is obtained in the python shell.



```
IDLE Shell 3.11.8
File Edit Shell Debug Options Window Help

Data: a31.30b87.00c89.94d84e

Room Temperature: 31.3
Room Humidity: 87.0
Body Temperature: 89.94
Heart Beat: 84
Uploading data to webserver....
Data Successfully Uploaded!

Predictions:
X Prediction: 0
Y Prediction: 1
Z Prediction: 0
w Prediction: 1
Abnormal Temperature Detected
ds18b20: 89.94 ( 89.94 )
Abnormal Body Temperature Detected
heart_beat: 1 ( 84 )
Heart Beat is Normal

-----
      == Data Received ==
-----

Data: a31.30b87.00c89.94d102e
```

Fig:5.6 Output in python shell.

The first line of the output represents the data that is provided from the arduino board in the form of a single string. The string can be split to get our required readings from the sensor.

- a to b is for the room temperature.
- b to c is for the room humidity.
- c to d is for the body temperature.
- d to e is for the heartbeat.

After splitting the values from the string the individual readings are shown in the python shell below the data line. These values are separated by the new line in order to avoid the unnecessary confusion and to improve the accuracy while reading the data.

Table:5.1 Collected data in tabular form

SI NO.	Room temperature	Humidity	Body temperature	Heart beat
1	29.5	54	90	105
2	30.5	37	90	94
3	30.5	50	89	84
4	29.3	38	85	85
5	26.2	36	87	110

After the successful printing of the data the next rows shows the predictions values of each data. Here the X, Y, Z, W are nothing but the individual prediction values of each data room temperature, room humidity, body temperature and heartbeat.

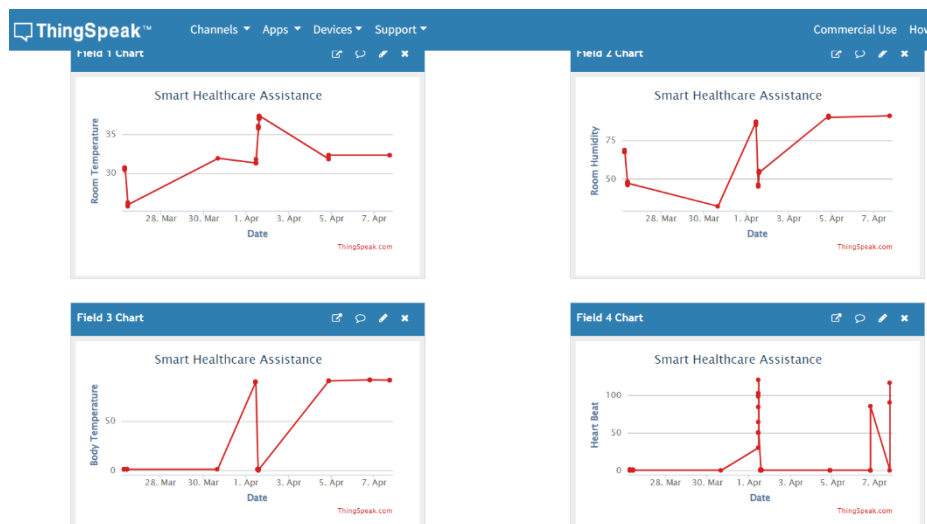


Fig:5.7 Output in thingspeak website.

After the getting the output in the python ide simultaneously the data is uploaded to the thingspeak cloud platform. Here the user can monitor and store the his medical data.

```

Predictions:
X Prediction: 0
Y Prediction: 1
Z Prediction: 0
w Prediction: 1
Abnormal Temperature Detected
ds18b20: 89.38 ( 89.38 )

```

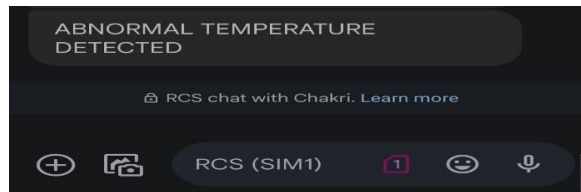


Fig:5.8 Sending alert to the user (abnormal temperature) through GSM.

If there is any abnormality in the readings then the GSM send the alert to the mobile phone in the form of the text message.

Table:5.2 Performance table.

SI NO	SAMPLE SIZE	ACCURACY
1	45%	0.95
2	50%	0.94
3	70%	0.94
4	80%	0.90

```

from sklearn.metrics import classification_report, accuracy_score
print(classification_report(y_test, y_pred))

```

```

              precision    recall  f1-score   support

0               0.92        0.98        0.95         60
1               0.97        0.88        0.92         40

 accuracy                   0.94         100

```

Fig:5.9 Test data size accuracy test.

By using the above data accuracy score can be calculated. It is calculated as ratio of sum of total number of correctly predicted to the total sum of correctly predicted and wrongly predicted. To obtain confusion matrix confusion_matrix function is used. Accuracy_score is function used to calculate accuracy. Here 94 percent of accuracy is obtained. Precision of the 1's is 97 percent which tells percentage of the correct predictions to the total positive predictions. Similarly, precision for 0's is 92 percent.

CHAPTER 6

CONCLUSION

It is necessary to maintain the balanced health. Maintaining balanced health is everyone's aim. Continuous monitoring has been increasing a lot day-by-day. The prototype of implementation of IoT and AI based system for smart healthcare assistance can make the user more conscious about their health, reacting by sending messages for some particular changes happened. This will give the collection of data of particular surroundings of its placement. The storage of data collected by sensors in the IoT platform is very useful for performing analysis on data to get next state of output. By analysing the present state and the data sheet values the predicted output is obtained using machine learning algorithm helps in taking care about the health. It is difficult to bring the technology into healthcare using AI and IoT rapidly. This project provides a small idea to prevent any health instability before it occurs (Because Prevention is better than cure). The project provides a prototype or an idea for smart healthcare assistance and continuous monitoring of changes in body temperature and surrounding temperature, heart beat which helps prevention of health instabilities.

REFERENCES:

- [1] C. Graham, "Fear of the unknown with healthcare IoT devices: An exploratory study," *Inf. Secur. J. Glob. Perspect.*, vol. 30, no. 2, pp. 100–110, 2021.
- [2] F. Wu, C. Qiu, T. Wu, and M. R. Yuce, "Edge-based hybrid system implementation for long-range safety and healthcare IoT Applications," *IEEE Internet Things J.*, vol. 8, no. 12, pp. 9970–9980, 2021.
- [3] Hina Sattar, Imran Sarwar Bajwa, Nadeem Sarwar, Umar Shafi, Noreen Jamil, M.G Abbas Malik "An IoT-based Intelligent Wound Monitoring System". Digital Object Identifier 10.1109/ACCESS.2019.
- [4] J. Chen, A. Valehi, and A. Razi, "Smart heart monitoring: Early prediction of heart problems through predictive analysis of ECG signals," *IEEE Access*, vol. 7, pp. 120831– 120839, 2019.
- [5]. K Murali Krishna, Yogini Dilip Borole, Smita Sharma, Surendra Pal Singh, Santhosh Kumar R, Anurag Shrivastava "Smart Robotic System for Remote Health Monitoring of the Environment Using IoT and Global Positioning System". 2021 International Conference on Technological Advancements and Innovations (ICTAI) | 978-1-6654-2087-7/21/\$31.00 ©2021 IEEE 10.1109/ICTAI53825.2021.9673276.
- [6] L. Chen, J. Hoey, C.D. Nugent, D.J. Cook, Z. Yu, Sensorbased activity recognition. *IEEE Trans. Syst. Man Cybern. C Appl. Rev.*, vol. 99, pp.1–19, 2012.
- [7] L. Greco, G. Percannella, P. Ritrovato, F. Tortorella, and M. Vento, "Trends in IoT based solutions for health care: Moving AI to the edge," *Pattern Recognit. Lett.*, vol. 135, pp. 346–353, 2020.
- [8]. Md Julhas hossain, Md. Amdadul Bari and Mohammad Monirujjaman Khan "Development of an IoT Based Health Monitoring System for e-Health" 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC) | 978-1-6654-8303-2/22/\$31.00 ©2022 IEEE 10.1109/CCWC54503.2022.9720825.
- [9] M. Javaid and I. H. Khan, "Internet of Things (IoT) enabled healthcare helps to take the challenges of COVID-19 Pandemic," *J. Oral Biol. Craniofac. Res.*, vol. 11, no. 2, pp. 209–214, 2021.

- [10]. Mutra Venkata Sai Girish, Azaad Pallam, P Divyashree, Agraj Khare, Priyanka Dwivedi "IoT Enabled Smart Healthcare Assistance for Early Prediction of Health Abnormality". 2021 IEEE International Symposium on Smart Electronic Systems (iSES) 978-1-7281-8753-2/21/\$31.00 ©2021 IEEE | DOI 10.1109/iSES52644.2021.
- [11] P. Sundaravadivel, E. Kougianos, S. P. Mohanty, and M. K. Ganapathiraju, "Everything you wanted to know about smart health care: Evaluating the different technologies and components of the internet of things for better health," IEEE consum. electron. mag., vol. 7, no. 1, pp. 18–28, 2018.
- [12] R. Rajavel, S. K. Ravichandran, K. Harimoorthy, P. Nagappan, and K. R. Gobichettipalayam, "IoT-based smart healthcare video surveillance system using edge computing," J. Ambient Intell. Humaniz. Comput., 2021.
- [13]. RoyChoudhury, S., Umasankar, Y., Jaller, J., Herskovitz, I., Mervis, J., Darwin, E., ... & Bhansali, S. (2018). Continuous Monitoring of Wound Healing Using a Wearable Enzymatic Uric Acid Biosensor. Journal of The Electrochemical Society, 165(8), B3168-B3175.
- [14]. Santhi, V., Ramya, K., Tarana, A. P. J., & Vinitha, G. (2017). IOT based wearable health monitoring system for pregnant ladies using cc3200. International Journal of Advanced Research Methodology in Engineering & Technology, 1, 3.
- [15] S. K. Routray and S. Anand, "Narrowband IoT for healthcare," in 2017 International Conference on Information Communication and Embedded Systems (ICICES), pp. 1–4, 2017.
- [16]. Sonali Mahajan, A.M.Birajdar "IOT based Smart Health Monitoring System for Chronic Diseases". 2019 IEEE Pune Section International Conference (PuneCon).978-1-7281-1924-3/19/\$31.00 ©2019 IEEE.
- [17] S. Satpathy, P. Mohan, S. Das, and S. Debbarma, "A new healthcare diagnosis system using an IoT-based fuzzy classifier with FPGA," J. Supercomput., vol. 76, no. 8, pp. 5849–5861, 2020.
- [18]. Tamanna Shaown, Imam Hasan, Md.Muradur Rahman Mim, Dr. Md. Shohrab Hossain "IoT-based Portable ECG Monitoring System for Smart Healthcare". 978-1-7281-3445-1/19/\$31.00 ©2019 IEEE.

[19]. Vinayagam P, Hameem Irfana A, Sireesha K, Tabassum Fathima N, Tanuja K "An AI Tool for Emergency Medical Assistance System". 2022 3rd International Conference on Smart Electronics and Communication (ICOSEC) | 978-1-6654-9764-0/22/\$31.00 ©2022 IEEE | DOI: 10.1109/ICOSEC54921.2022.9951990.

[20] Wan, J., A. A. H. Al-awlaqi, M., Li, M. et al. Wearable IoT enabled real-time health monitoring system. J Wireless Com Network, vol. 298, pp .1-10, 2018.