SMART HEALTH MONITORING SYSTEM

MINI-PROJECT REPORT

Submitted by

VISHNU.S	**200701295**
YOKESH.R	**200701303**
YUVARAJA.M	**200701305**

In partial fulfilment of the award of the degree

of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE

RAJALAKSHMI ENGINEERING COLLEGE, CHENNAI

MAY 2023

BONAFIDE CERTIFICATE

Certified that this project **"SMART HEALTH MONITORING SYSTEM"** is the bonafide work of **"VISHNU.S, YOKESH.R, YUVARAJA.M"** who carried out the project work under my supervision.

Dr.S. ANANTHA SIVAPRAKASAM

ACADEMIC HEAD

ASSISTANT PROFESSOR

Dept. of Computer Science and Engg,

Rajalakshmi Engineering College,

Chennai.

Mrs. SUSMITA MISHRA

ASSISTANT PROFESSOR

Dept. of Computer Science and Engg,

Rajalakshmi Engineering College,

Chennai.

This mini-project report is submitted for the viva vice examination to be held on ____________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

I would like to express my special thanks of gratitude to my teacher “**Mrs. SUSMITA MISHRA**” for her able guidance and support in completing our project.

I would also like to extend my gratitude to my HOD “**DR. P. KUMAR**” for providing us with all the facilities that was required.

VISHNU.S

YUVARAJA.M

YOKESH.R

ABSTRACT

Health monitoring systems have rapidly evolved during the past two decades and have the potential to change the way health care is currently delivered. Although smart health monitoring systems automate patient monitoring tasks and, thereby improve the patient workflow management, their efficiency in clinical settings is still debatable. This project is on IOT device which is used to calculate the human blood pressure and heart rate. The Internet of Things (IoT) has enabled the invention of smart health monitoring systems. These health monitoring systems can track a person's mental and physical wellness. This can be achieved using Arduino kit, sensors and actuators. It can help patients in the case of emergency by providing immediate health consultation from the doctor available at a distant location based smart systems enable remote monitoring of the patient by the guardian/ family member which is considered as one of the major advantages to save the precious human life.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

The healthcare industry is constantly seeking ways to improve patient care, particularly through the integration of technology. One of the most significant technological advancements in healthcare is the Internet of Things (IoT), which involves the connection of everyday devices to the internet for data collection and analysis. IoT devices have the potential to revolutionize the healthcare industry by providing real-time monitoring of patients' health, allowing doctors to make more informed decisions about patient care.

In this project, we have developed an IoT-based healthcare system that utilizes a heart rate sensor and an ESP8266 microcontroller to monitor patients' heart rate remotely. The project is designed to provide a cost-effective and convenient solution for patients to monitor their heart rate in real-time, from the comfort of their own homes. The heart rate sensor is a non-invasive device that is placed on the patient's wrist, making it easy and comfortable to use. The ESP8266 microcontroller is programmed to read the heart rate data from the sensor and transmit it to Adafruit.io, a cloud-based IoT platform, for storage and analysis.

Heart rate monitoring is an essential aspect of healthcare, particularly for patients with heart conditions. Real-time monitoring of the heart rate allows doctors to detect any irregularities or anomalies, which can be indicative of an underlying medical condition.

Remote monitoring of heart rate is particularly useful for patients who are unable to visit a hospital or clinic frequently. The IoT-based healthcare system developed in this project provides a convenient and cost-effective solution for remote heart rate monitoring.

The heart rate sensor used in this project is a photoplethysmography (PPG) sensor, which measures the changes in blood volume in the capillaries under the skin. The sensor data is then converted into an electrical signal, which is processed by the ESP8266 microcontroller. The microcontroller uses Wi-Fi to send the heart rate data to Adafruit.io, where it is stored and analyzed. Patients can access their heart rate data from any device that has an internet connection, such as a smartphone or a laptop. The data is presented in the form of graphs and charts, which allow patients to monitor their heart rate patterns over time.

In conclusion, the healthcare system developed in this project offers a convenient and cost-effective solution for remote heart rate monitoring. The integration of IoT technology with healthcare has the potential to revolutionize the industry, providing patients with real-time monitoring of their health and doctors with more informed decisions about patient care. This project serves as a proof-of-concept for the use of IoT technology in healthcare and provides a foundation for further research and development in the field.

CHAPTER 2

LITERATURE SURVEY

The integration of Internet of Things (IoT) technology with healthcare has the potential to revolutionize the industry, particularly in the area of remote patient monitoring. The use of IoT devices for remote monitoring of patients' health can provide real-time data, allowing doctors to make more informed decisions about patient care. In this literature survey, we review several studies that have investigated the use of IoT-based heart rate sensors for remote monitoring of patients' heart rate.

In a study published in the Journal of Medical Systems, researchers investigated the use of an IoT-based heart rate sensor for remote monitoring of patients with heart failure. The study involved 30 patients who were provided with an IoT-based heart rate sensor that transmitted their heart rate data to a cloud-based platform for storage and analysis. The study found that the use of the IoT-based heart rate sensor improved the accuracy and reliability of heart rate monitoring, and allowed doctors to detect any irregularities or anomalies in real-time.

In a study published in the Journal of Healthcare Engineering, researchers investigated the use of an IoT-based heart rate sensor for remote monitoring of patients with chronic obstructive pulmonary disease (COPD). The study involved 20 patients who were provided with an IoT-based heart rate sensor that transmitted their heart rate data to a cloud-based platform for storage and analysis. The study found that the use of the IoT-based heart rate sensor improved the management of COPD, by providing doctors with real-time data on the patients' heart rate patterns.

In a study published in the Journal of Medical Internet Research, researchers investigated the use of an IoT-based heart rate sensor for remote monitoring of patients with atrial fibrillation (AF). The study involved 50 patients who were provided with an IoT-based heart rate sensor that transmitted their heart rate data to a cloud-based platform for storage and analysis. The study found that the use of the IoT-based heart rate sensor improved the management of AF, by providing doctors with real-time data on the patients' heart rate patterns and allowing them to adjust their medication and treatment plans accordingly.

The studies reviewed in this literature survey demonstrate the potential of IoT-based heart rate sensors for remote monitoring of patients' heart rate. The use of IoT technology in healthcare can provide real-time data on patients' health, allowing doctors to make more informed decisions about patient care. The integration of IoT technology with heart rate sensors can improve the accuracy and reliability of heart rate monitoring, particularly for patients with heart conditions.

EXISTING SYSTEM

Existing systems for remote heart rate monitoring using IoT technology vary in their design, features, and applications. Some of the popular systems available in the market includes Fitbit is a popular wearable device that tracks physical activity, heart rate, and sleep patterns. The device uses sensors to monitor heart rate and sends the data to a mobile app for analysis. Apple Watch is a smartwatch that includes a heart rate monitor, which can be used to detect irregular heart rhythms. The watch can send alerts to the user if the heart rate exceeds a certain threshold. AliveCor Karda is a portable electrocardiogram (ECG) device that attaches to a smartphone and allows users to capture ECG readings in real-time. The device is designed to detect abnormal heart rhythms and send the data to a cloud-based platform for analysis. In existing project they have made some app integration and used some other sensors to detect heart rate.

CHAPTER 3

PROJECT DESCRIPTION

3.1 PROPOSED SYSTEM

Our proposed system is an IoT-based healthcare solution that utilizes a heart rate sensor to monitor patients' heart rate remotely. The system consists of two main components: the heart rate sensor and a cloud-based platform for storage and analysis of patient data.The heart rate sensor is an IoT device that can be attached to the patient's body, and it continuously measures the patient's heart rate. We will be using an ESP8266 microcontroller along with a heart rate sensor to achieve this. The ESP8266 is a low-cost Wi-Fi chip that allows the sensor to connect to the internet, making it possible to transmit the heart rate data to a cloud-based platform in real-time.The heart rate sensor will be connected to the patient's body through a comfortable wearable device, which can be worn around the wrist or attached to the chest. The wearable device will be designed to ensure patient comfort, allowing them to continue their daily activities without any discomfort.

The heart rate data collected by the sensor will be transmitted to a cloud-based platform for storage and analysis. We will be using the Adafruit.io platform for this project as it provides an easy-to-use interface for receiving data and storing it in the cloud. The cloud-based platform will store the patient's heart rate data and provide real-time data analysis.

The platform will also be designed to send alerts to doctors or caregivers if the patient's heart rate exceeds a certain threshold, indicating a potential health issue. Additionally, the platform will be equipped with machine learning algorithms that can analyze heart rate data over time, providing insights into the patient's health trends and detecting any anomalies or irregularities in the heart rate data.

The proposed system aims to provide doctors and caregivers with a cost-effective and convenient solution for remote heart rate monitoring, improving patient outcomes and quality of life. The use of IoT technology in healthcare can provide real-time data on patients' health, allowing doctors to make more informed decisions about patient care. The integration of IoT technology with heart rate sensors can improve the accuracy and reliability of heart rate monitoring, particularly for patients with heart conditions.

3.2 REQUIREMENTS

- ESP8266 Micro Controller
- Pulse sensor (MAX 30100)
- Buck converter
- Jumper Wires

Buck Converter:

The buck converter is a well-known and simple type of DC-DC converter. We used buck converters in our system because they produce an output voltage that is less than their input. This converter is utilized for proficient control transformation that amplifies battery life and decreases warmth. It shows the prototype of the buck converter

Figure 3.2.1 Buck Converter

ESP8266 Micro Controller

We used the node MCU ESP8266 for this system, which is a wireless module, because the ESP8266 microcontroller has Wi-Fi capability, and the node MCU has a wireless system that can send data to a server. The node MCU has an asynchronous receiver-transmitter serial communication module, which enables it to communicate with the Bluetooth module. The node MCU ESP8266 microcontroller can operate with a power supply of 3.3 V operating voltage and a 7 to 12 V input voltage. It has a flash memory of 4 Mb and an SRAM of 64 Kb. It has 16 digital input and output pins and one analog input pin. The node MCU also has a PCB antenna The node MCU wireless module sends the measured pulse rate, oxygen saturation, and temperature to the server. This component was chosen because it links the server IP address to the node MCU to obtain the measured value through a mobile application. The node MCU is an open-source Lua-based firmware and an advancement board. It is specially designed for IoT-based applications, and this component plays a vital role in our system.

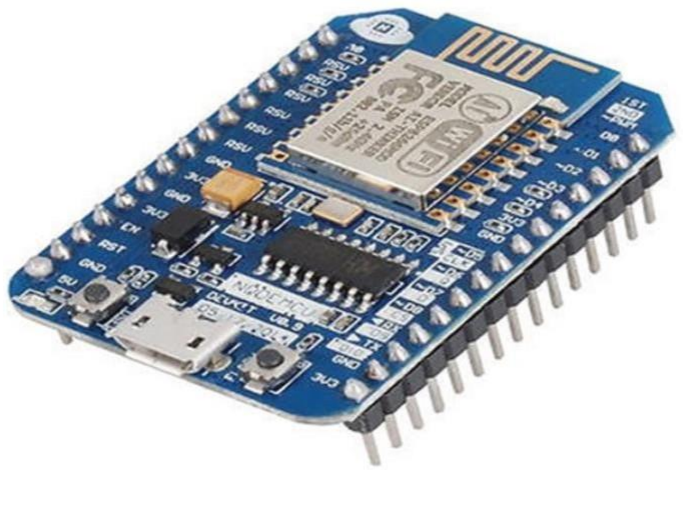


Figure 3.2.2 Esp8266 micro controller

Pulse sensor:

MAX30100 is a sensor that can measure blood oxygen saturation level and pulse rate. It shows the prototype of the SpO2 Pulse Sensor (MAX30100). Saturation of peripheral oxygen (SpO2) is a calculation of blood vessel oxygen saturation, which refers to the amount of oxygenated hemoglobin in the blood. In a human body, ordinary SpO2 values range from 90to 100%. In this system, a MAX 30100 pulse oximeter was suitable. It is a coordinated beat oximeter and heart rate sensor arrangement, which provides precise values. This sensor combines two LEDs, a photo detector, optimized optics, and low-noise analog flag handling to identify beat oximetry and heart rate signals; hence, it is suitable for this system.

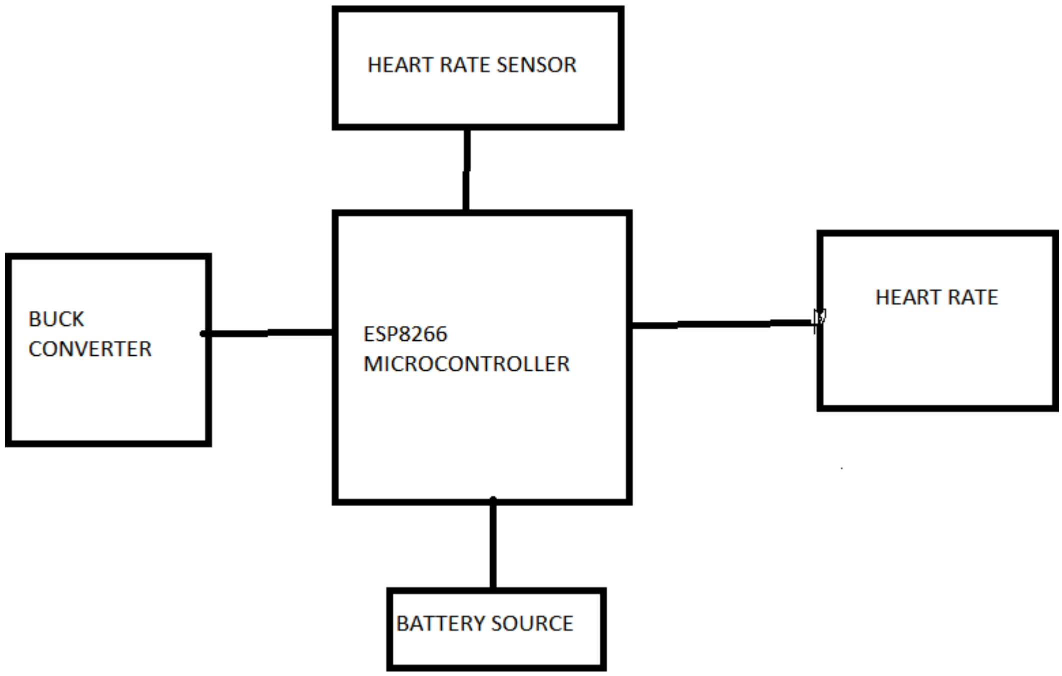
Figure 3.2.3 Pulse rate sensor

Jumper wires

The term "jumper wire" simply refers to a conducting wire that establishes an electrical connection between two points in a circuit. A jump wire is an electrical wire,or group of them in a cable, with a connector or pin at each end, which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

Figure 3.2.4 Jumper wires

3.3 SYSTEM ARCHITECTURE:

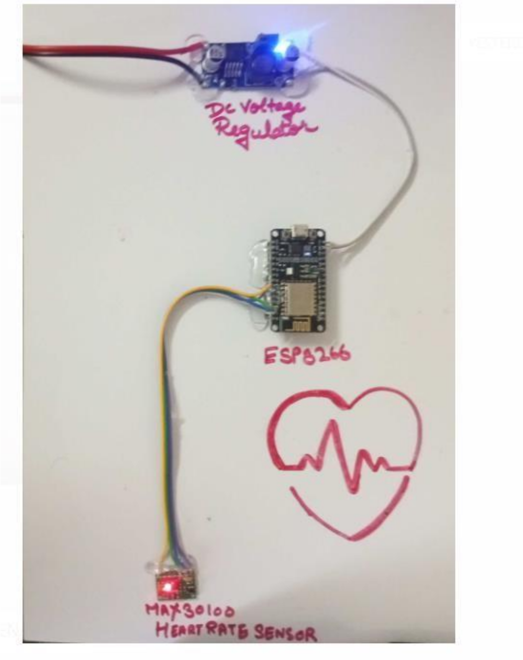


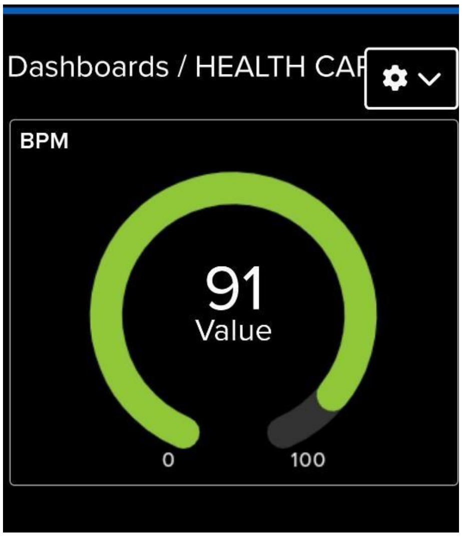
EXPLANATION

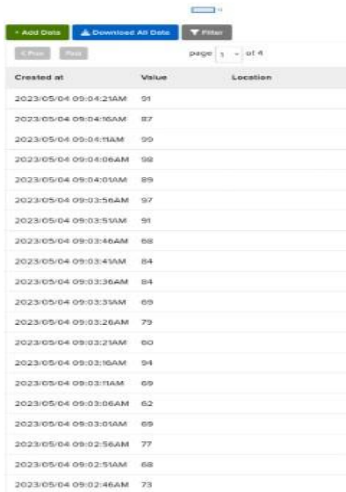
he project uses an ESP8266 microcontroller, a heart rate sensor, and Adafruit.io cloud platform to create an IoT-based healthcare system. The heart rate sensor is connected to the ESP8266 microcontroller, which is programmed to read the sensor data and transmit it to Adafruit.io via Wi-Fi. The sensor data is then stored on the Adafruit.io platform, which provides real-time visualization of the data in the form of graphs and charts. This allows patients to monitor their heart rate in real-time and track any changes in their heart rate patterns.

The heart rate sensor used in this project is a non-invasive sensor that can be placed on the skin of the patient's wrist. The sensor measures the changes in blood volume in the capillaries under the skin, which are caused by the contraction and relaxation of the heart muscles. The sensor data is then converted into an electrical signal, which is processed by the ESP8266 microcontroller. The microcontroller uses Wi-Fi to send the data to Adafruit.io, where it is stored and analyzed.The Adafruit.io platform provides various tools to analyze the heart rate data, such as graphs and charts that show the heart rate patterns over time. Patients can access the data from any device that has an internet connection, such as a smartphone or a laptop.

3.4 OUTPUT SCREENSHOTS







Created at	Value	Location
2023/05/04 09:04:21AM	91	
2023/05/04 09:04:16AM	87	
2023/05/04 09:04:11AM	99	
2023/05/04 09:04:06AM	98	
2023/05/04 09:04:01AM	89	
2023/05/04 09:03:56AM	97	
2023/05/04 09:03:51AM	91	
2023/05/04 09:03:46AM	68	
2023/05/04 09:03:41AM	84	
2023/05/04 09:03:36AM	84	
2023/05/04 09:03:31AM	69	
2023/05/04 09:03:26AM	79	
2023/05/04 09:03:21AM	60	
2023/05/04 09:03:16AM	94	
2023/05/04 09:03:11AM	69	
2023/05/04 09:03:06AM	62	
2023/05/04 09:03:01AM	69	
2023/05/04 09:02:56AM	77	
2023/05/04 09:02:51AM	68	
2023/05/04 09:02:46AM	73	

CODE:

```
#include <Wire.h>
#include "MAX30100_PulseOximeter.h"
#include <ESP8266WiFi.h>

#include  <ESP8266WiFi.h>
#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"

#define REPORTING_PERIOD_MS 1000
#ifdef ESP32
#pragma message(THIS EXAMPLE IS FOR ESP8266 ONLY!)
#error Select ESP8266 board.
#endif



// Connections : SCL PIN - D1 , SDA PIN - D2 , INT PIN - D0
PulseOximeter pox;

float BPM, SpO2;
uint32_t tsLastReport = 0;



/********* WiFi Access Point ***********/

#define WLAN_SSID       "YOKESH.R"
#define WLAN_PASS       "12345666"
#define AIO_SERVER      "io.adafruit.com"
#define AIO_SERVERPORT  1883
```

```
#define AIO_USERNAME    "YOKESH_8863"
#define AIO_KEY         "aio_xMKx57QyfZN2KCPnKfaxnEwGDhrJ"
WiFiClient client;
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT,
AIO_USERNAME, AIO_KEY);
Adafruit_MQTT_Publish photocell = Adafruit_MQTT_Publish(&mqtt,
AIO_USERNAME "/feeds/photocell");
Adafruit_MQTT_Subscribe onoffbutton = Adafruit_MQTT_Subscribe(&mqtt,
AIO_USERNAME "/feeds/onoff");
void MQTT_connect();




const unsigned char bitmap [] PROGMEM =
{
  0x00, 0x00, 0x00, 0x00, 0x01, 0x80, 0x18, 0x00, 0x0f, 0xe0, 0x7f, 0x00, 0x3f, 0xf9,
0xff, 0xc0,
  0x7f, 0xf9, 0xff, 0xc0, 0x7f, 0xff, 0xff, 0xe0, 0x7f, 0xff, 0xff, 0xe0, 0xff, 0xff, 0xff,
0xf0,
  0xff, 0xf7, 0xff, 0xf0, 0xff, 0xe7, 0xff, 0xf0, 0xff, 0xe7, 0xff, 0xf0, 0x7f, 0xdb, 0xff,
0xe0,
  0x7f, 0x9b, 0xff, 0xe0, 0x00, 0x3b, 0xc0, 0x00, 0x3f, 0xf9, 0x9f, 0xc0, 0x3f, 0xfd, 0xbf,
0xc0,
  0x1f, 0xfd, 0xbf, 0x80, 0x0f, 0xfd, 0x7f, 0x00, 0x07, 0xfe, 0x7e, 0x00, 0x03, 0xfe, 0xfc,
0x00,
  0x01, 0xff, 0xf8, 0x00, 0x00, 0xff, 0xf0, 0x00, 0x00, 0x7f, 0xe0, 0x00, 0x00, 0x3f, 0xc0,
0x00,
  0x00, 0x0f, 0x00, 0x00, 0x00, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00
```

```
};

void onBeatDetected()
{
  Serial.println("Beat Detected!");
}

void setup()
{
  Serial.begin(115200);
  Serial.print("Initializing Pulse Oximeter..");
  if (!pox.begin())
  {
    Serial.println("FAILED");
    for (;;);
  }
  else
  {
    Serial.println("SUCCESS");
    pox.setOnBeatDetectedCallback(onBeatDetected);
  }
  Serial.println(F("Adafruit MQTT demo"));
  Serial.println(); Serial.println();
  Serial.print("Connecting to ");
  Serial.println(WLAN_SSID);
  WiFi.begin(WLAN_SSID, WLAN_PASS);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
```

```
  Serial.println();

  Serial.println("WiFi connected");
  Serial.println("IP address: "); Serial.println(WiFi.localIP());
}

void loop()
{
  pox.update();
  BPM = pox.getHeartRate();
  SpO2 = pox.getSpO2();
  if (millis() - tsLastReport > REPORTING_PERIOD_MS)
  {
    Serial.print("Heart rate:");
    Serial.print(BPM);
    Serial.print(" SpO2:");
    Serial.print(SpO2);
    Serial.println(" %");
    //Blynk.virtualWrite(V0, pox.getHeartRate());
    //Blynk.virtualWrite(V3, dht.getHumidity());

    tsLastReport = millis();
  }
}

void MQTT_connect() {
  int8_t ret;

  // Stop if already connected.
  if (mqtt.connected()) {
```

```
    return;
  }
  Serial.print("Connecting to MQTT... ");
  uint8_t retries = 3;
  while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected
    Serial.println(mqtt.connectErrorString(ret));
    Serial.println("Retrying MQTT connection in 5 seconds...");
    mqtt.disconnect();
    delay(5000); // wait 5 seconds
    retries--;
    if (retries == 0) {
      // basically die and wait for WDT to reset me
      while (1);
    }
  }
  Serial.println("MQTT Connected!");
}
```

CHAPTER 4

CONCLUSION AND FUTURE WORK

In conclusion, the healthcare system developed in this project offers a convenient and cost-effective solution for remote heart rate monitoring. The integration of IoT technology with healthcare has the potential to revolutionize the industry, providing patients with real-time monitoring of their health and doctors with more informed decisions about patient care. This project serves as a proof-of-concept for the use of IoT technology in healthcare and provides a foundation for further research and development in the field. The proposed system has several future scope for improvement and expansion, such as Integration with other sensors: In addition to heart rate sensors, the proposed system can be integrated with other sensors, such as blood pressure sensors, temperature sensors, and oxygen saturation sensors, to provide a more comprehensive health monitoring system. Personalized health insights: The proposed system can be further enhanced by using machine learning algorithms to provide personalized health insights to patients, such as diet and exercise recommendations, based on their heart rate data. Integration with electronic health records (EHRs): The proposed system can be integrated with electronic health records (EHRs) to provide doctors with a more comprehensive view of patients' health data and improve the accuracy of diagnosis and treatment.

CHAPTER 5

REFERENCES

Research Papers

1) Remote Health Monitoring Using IoT-Based Sensors: A Comprehensive Review" by Bhagya Nathali Silva, Sasika Manjula, R. M. A. P. Rajapakse, and Anuradha Karunarathne: This paper provides a comprehensive review of remote health monitoring systems that utilize IoT-based sensors, including those that monitor heart rate.

2) "IoT Based Heart Rate Monitoring System for Healthcare Applications" by S. Srinivasan, S. Radhakrishnan, and M. Thirumaran: This paper presents an IoT-based heart rate monitoring system that uses a wearable device to monitor heart rate and transmits the data to a mobile app for analysis.

3)"Design and Implementation of an IoT-Based Wearable Heart Rate Monitoring System" by Yuanyuan Yang, Xiaoping Wang, and Meng Zhang: This paper presents a wearable heart rate monitoring system that uses an IoT-based architecture to collect and analyze heart rate data.

4)"Remote Heart Rate Monitoring System Based on IoT and Cloud Computing" by Jiahui Zeng, Jie Chen, and Hongming Cai

Web-Links

1) https://www.robotique.tech/robotics/health monitoring -system-with-arduino/
2) https://www.electroniclinic.com/heart rate -sensor-obstacle-using-arduino-hc-sr04/
3) https://steemit.com/utopian-io/@ted7/arduinoIDE
4) https://www.hindawi.com/journals/cmmm/2021/8591036/