Arduino Powered Health Tracker

Project Report

Submitted for the partial fulfilment of the degree of

Bachelor of Technology

In

Internet of Things (IOT)

Submitted By

Princy Gupta (0901IO221055)

UNDER THE SUPERVISION AND GUIDANCE OF

Dr. Aftab Ahmed Ansari
Assistant Professor

Centre for Internet of Things

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR (M.P.), INDIA
माधव प्रौद्योगिकी एवं विज्ञान संस्थान, ग्वालियर (म.प्र.), भारत
Deemed University
NAAC ACCREDITED WITH A++ GRADEA

Jul-Dec 2024

DECLARATION BY THE CANDIDATE

We hereby declare that the work entitled "Arduino Powered Health Tracker" is our work, conducted under the supervision of **Dr.Aftab Ahmed Ansari, Assistant Professor** , during the session Jul-Dec 2024. The report submitted by us is a record of bonafide work carried out by us.

We further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Princy Gupta
0901IO221055
B.Tech. V Sem

Date: 19th Nov,2024
Place: Gwalior

This is to certify that the above statement made by the candidates is correct to the best of my knowledge and belief.

Guided By:

Dr. Aftab Ahmed Ansari
Assistant Professor
Center for IOT
MITS, Gwalior

Departmental Project Coordinator

Dr. Nookala Venu
Assistant Professor
Centre for Internet of Things
MITS, Gwalior

Approved by HoD

Dr. Praveen Bansal
Assistant Professor
Centre for Internet of Things
MITS, Gwalior

PLAGIARISM CHECK CERTIFICATE

This is to certify that we, students of B.Tech. in **Internet of Things (IOT)** have checked our complete report entitled "**Arduino Powered Health Tracker**" for similarity/plagiarism using the "Duplichecker" software available in the institute.

This is to certify that the similarity in our report is found to be 15 % which is within the specified limit (20%).

The full plagiarism report along with the summary is enclosed.

Prupta

Princy Gupta
0901IO221055
B.Tech. V Sem

Checked & Approved By:

Soumyajit Ghosh 19/11/24

Dr. Soumyajit Ghosh
Assistant Professor
Centre for Internet of Things
MITS, Gwalior

ACKNOWLEDGEMENT

The full semester Project has proved to be pivotal to our career. We are thankful to our institute, **Madhav Institute of Technology & Science** to allow us to continue our disciplinary/interdisciplinary Project as a curriculum requirement, under the provisions of the Flexible Curriculum Scheme approved by the Academic Council of the institute. We extend our gratitude to the Director of the institute, **Dr. R. K. Pandit** and Dean Academics, **Dr. Manjaree Pandit** for this.

We would sincerely like to thank our department, **Centre for Internet of Things,** for allowing us to explore this project. We humbly thank **Dr. Praveen Bansal**, Assistant Professor and Coordinator, Centre for Internet of Things, for his continued support during the course of this engagement, which eased the process and formalities involved. We are sincerely thankful to our faculty mentors. We are grateful to the guidance of **Dr. Aftab Ahmed Ansari**, Assistant Professor, and Centre for Internet of Things, for his continued support and guidance throughout the project. We are also very thankful to the faculty and staff of the department.

Princy Gupta
0901IO221055
B.Tech. V Sem
Centre for Internet of Things

PAPERNAME
PrincyGupta.pdf

AUTHOR
SG

WORDCOUNT
3138Words

CHARACTERCOUNT
20019Characters

PAGECOUN
T
21Pages

FILESIZE
760.5KB

SUBMISSIONDATE
Nov18,20247:47PMGMT+5:30

REPORTDATE
Nov18,20247:48PMGMT+5:30

- 15%OverallSimilarity

ecombinedtotalofallmatches,includingoverlappingsources,foreachdatabase.

- 9%Internetdatabase
- 1%Publicationsdatabase
- Crossrefdatabase
- CrossrefPostedContentdatabas
- e13%SubmittedWorksdatabase

ExcludedfromSimilarityReport

- Bibliographicmaterial
- SmallMatches(Lessthen10words)

15% Overall Similarity

Top sources found in the following databases:

- 9% Internet database
- Crossref database
- 13% Submitted Works database
- 1% Publications database
- Crossref Posted Content database

TOP SOURCES

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	mitsgwalior on 2024-11-16 Submitted works	6%
2	mitsgwalior on 2024-05-27 Submitted works	3%
3	web.mitsgwalior.in Internet	2%
4	ijraset.com Internet	<1%
5	construction.jisc.ac.uk Internet	<1%
6	University of Central Lancashire on 2024-04-15 Submitted works	<1%
7	indianexpress.com Internet	<1%
8	Liberty University on 2024-06-09 Submitted works	<1%

9	**Mahagave Somnath, D Dhanalakshmi, Vuppulapu Sai Sripath, Shanmu...** Crossref	<1%
10	**Somaiya Vidyavihar on 2023-05-06** Submitted works	<1%
11	**Sotiris Athanaselis, Artemis Dona, Stavroula Papadodima, Giannis Pap...** Crossref	<1%
12	**dynlab.mpe.nus.edu.sg** Internet	<1%
13	**ijircce.com** Internet	<1%
14	**ijisae.org** Internet	<1%
15	**kscst.iisc.ernet.in** Internet	<1%

ABSTRACT

Arduino Powered Health Tracker

This project presents the development of a low-cost, portable health monitoring system utilizing the Arduino platform. The system aims to track key health parameters, including heart rate, blood oxygen saturation (SpO2), and body temperature, enabling individuals to monitor their well-being and detect potential health issues in real time.

The system comprises an Arduino microcontroller, a pulse oximeter sensor (MAX30102), a temperature sensor (DHT11), and a wireless communication module (ESP8266 or Bluetooth). The pulse oximeter sensor measures heart rate and SpO2 levels by emitting red and infrared light and analyzing the reflected light intensity. The temperature sensor measures ambient temperature, which can provide insights into potential fevers or other temperature-related health conditions. The wireless communication module enables data transmission to a mobile app or cloud-based platform for remote monitoring and analysis.

The Arduino microcontroller processes the sensor data, performs data filtering and signal processing techniques to improve accuracy, and transmits the processed data wirelessly. The mobile app or cloud platform displays the real-time health data, generates personalized health reports, and provides alerts for abnormal readings or potential health risks.

The system has been designed with user-friendliness and portability in mind. The compact and lightweight design allows for easy wearability, and the intuitive mobile app interface facilitates seamless data monitoring and analysis. The system's accuracy and reliability have been validated through rigorous testing and calibration procedures.

This Arduino-powered health tracker offers a cost-effective and accessible solution for individuals to proactively monitor their health and well-being. By empowering individuals with real-time health data, this system can contribute to early detection of health issues and promote preventive healthcare practices.

CONTENT

Table of Contents

NOMENCLATURE

General Technology Terms:

- **Arduino:** A popular open-source electronics platform used for building a variety of projects.
- **Microcontroller:** A small computer on a single integrated circuit.
- **Sensor:** A device that detects and responds to a physical stimulus.
- **LED:** Light-Emitting Diode, a semiconductor device that emits light when current flows through it.
- **Wireless Module:** Enables communication with other devices, such as smartphones or computers, over wireless networks like Wi-Fi or Bluetooth.
- **Firmware:** Software embedded in hardware devices, controlling their basic functions.
- **Programming Language:** The language used to write code for the microcontroller. Arduino typically uses a simplified version of C++.
- **IDE (Integrated Development Environment):** A software application that provides tools for writing, testing, and debugging code. Arduino IDE is a popular choice for Arduino projects.
- **Libraries:** Collections of pre-written code that can be used to perform specific tasks, such as reading sensor data or sending data over Wi-Fi.

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

This project aims to develop a low-cost, portable health monitoring system using an Arduino microcontroller. This device is designed to track vital health parameters such as heart rate, blood oxygen saturation (SpO2), and body temperature. By leveraging the power of IoT and cloud-based technologies, this system enables real-time monitoring, data analysis, and timely alerts to promote proactive healthcare.

The system consists of an Arduino board, various sensors, and a wireless module. The Arduino processes the sensor data, which is then transmitted to a cloud-based platform like ThingSpeak. This platform stores and visualizes the data, allowing users to monitor their health trends and receive notifications for abnormal readings.

The project's primary goal is to empower individuals to take control of their health by providing accessible and affordable health monitoring tools. By utilizing open-source hardware and software, this system offers a flexible and customizable solution for personal health management.

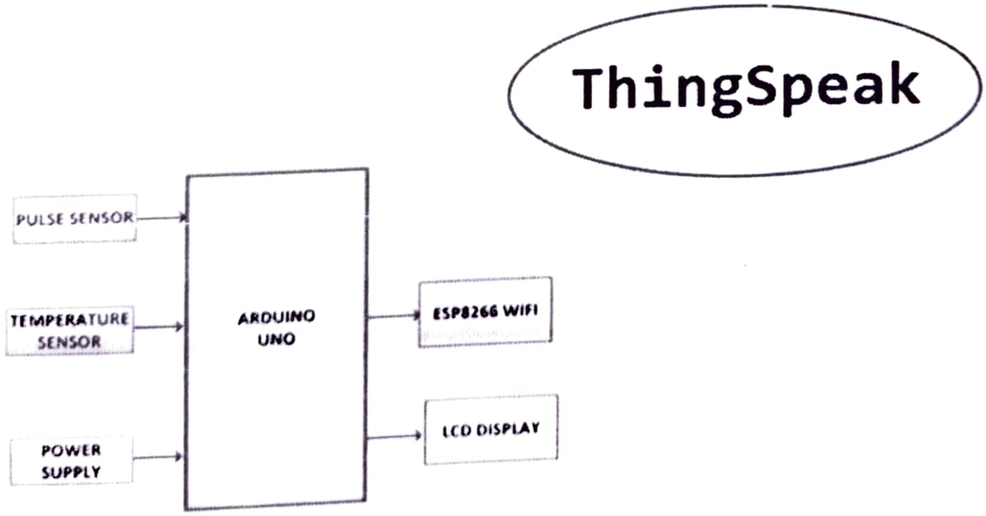

Fig.1 Basic Diagram

CHAPTER 2: LITERATURE SURVEY

The increasing prevalence of chronic diseases and the desire for personalized healthcare have driven the development of wearable health monitoring devices. These devices, ranging from simple fitness trackers to advanced medical-grade monitors, offer the potential to revolutionize healthcare by enabling continuous and remote monitoring of vital health parameters. While commercial options exist, they often come with high costs and limited customization options.

To address these limitations, there has been a growing interest in developing low-cost, open-source health monitoring systems based on microcontrollers like Arduino. Arduino's simplicity, versatility, and large community support make it an ideal platform for building custom health monitoring devices. Numerous research papers and projects have explored the use of Arduino for various health monitoring applications, including heart rate monitoring, blood pressure measurement, blood glucose level monitoring, and more.

One of the key challenges in developing Arduino-based health monitoring systems is ensuring the accuracy and reliability of sensor data. Several studies have investigated techniques for calibrating and validating sensors, as well as implementing signal processing algorithms to improve data quality. Additionally, the integration of wireless communication technologies, such as Wi-Fi and Bluetooth, enables seamless data transmission to smartphones or cloud-based platforms for remote monitoring and analysis.

While Arduino-based health monitoring systems offer a promising approach to personalized healthcare, it is essential to address potential limitations and challenges, such as battery life, sensor accuracy, and data security. Future research should focus on improving the design, performance, and user experience of these devices, as well as exploring innovative applications and integrations with other healthcare technologies.

CHAPTER 3: HARDWARE TOOLS

1. Pulse Sensor: The Pulse Sensor is a simple-to-use heart rate sensor compatible with Arduino. It incorporates an optical amplifier and noise reduction circuitry, allowing for accurate heart rate readings. By clipping the sensor to your earlobe or fingertip and connecting it to an Arduino, you can easily obtain real-time heart rate data. The sensor comes with pre-written Arduino code to facilitate its integration into various projects, making it accessible to students, artists, athletes, and developers.

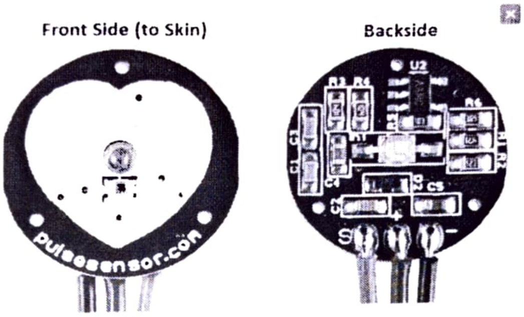

Fig.2 Pulse Sensor

2 LM35 Temperature Sensor: The LM35 series are precise integrated circuits that output a voltage directly proportional to the temperature in Celsius. They require no external calibration and offer high accuracy, typically within ±0.25°C at room temperature and ±0.75°C over a wide temperature range of -55°C to 150°C.

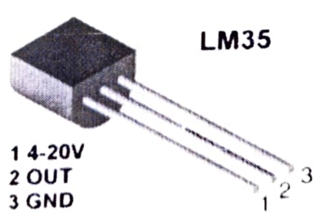

Fig.3 LM35 Sensor

3. ESP8266: The ESP8266 is a cost-effective and user-friendly device that can connect your projects to the internet. It can function as both a Wi-Fi access point and a station, allowing it to easily upload data to the internet and fetch data from online APIs. This makes it a versatile tool for IoT projects, enabling them to access and process information from the web.

Additionally, the ESP8266 can be programmed using the Arduino IDE, simplifying the development process for many users.

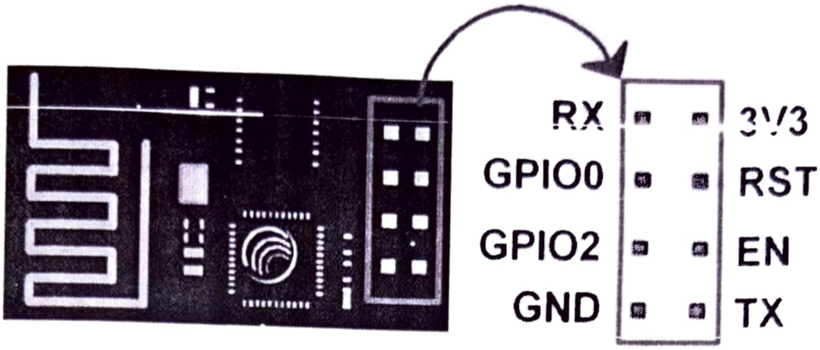

Fig.4 ESP8266

4. 16*2 LCD Display: A 16x2 LCD display is a common component used in various electronic projects. It consists of a 16-character by 2-line display, allowing for clear and concise information presentation. To interface an LCD with an Arduino, you typically need to connect it to the microcontroller's digital pins. The specific pins used will depend on the LCD model and your circuit design.

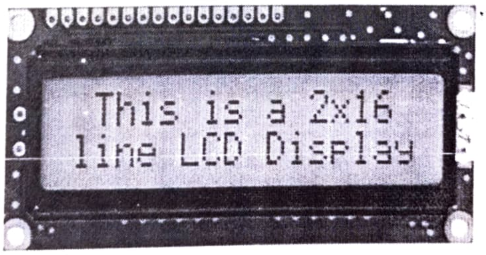

Fig.5 LCD Display

5. Jumper Wires: Jumper wires are straightforward electrical cables with connector pins on both ends that are used to connect parts of a microcontroller or a breadboard temporarily. Depending on the needs of the connection, they can be classified as male-to-male, female-to-female, or male-to-female. The sensors (such as the MQ-3 or IR sensor) in an IoT helmet are connected to the NodeMCU via jumper wires, which facilitates seamless data transfer between the parts.

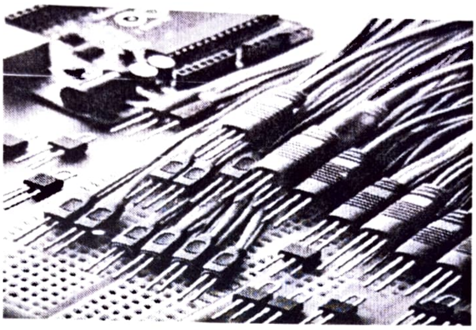

Fig.6 Jumper Wires

CHAPTER 4: METHODOLOGY AND ALGORITHM

METHODOLOGY

Hardware Selection and Setup:

1. **Microcontroller:** An Arduino Uno was chosen due to its simplicity and widespread availability.
2. **Sensors:** A pulse oximeter sensor was selected for heart rate and SpO2 measurement, and a temperature sensor (LM35) was chosen for temperature monitoring.
3. **Wireless Module:** An ESP8266 module was selected for Wi-Fi connectivity, enabling data transmission to the cloud.
4. **Display:** A 16x2 LCD display was used to provide real-time feedback on the measured parameters.

Software Development:

1. **Arduino IDE:** The Arduino IDE was used to write the code for the microcontroller.
2. **Sensor Libraries:** Libraries for the pulse oximeter and temperature sensors were integrated to facilitate data acquisition.
3. **Wi-Fi Connectivity:** The ESP8266 module was configured to establish a Wi-Fi connection and transmit data to the ThingSpeak cloud platform.
4. **LCD Display:** The LCD was programmed to display the measured heart rate, SpO2, and temperature values.

Cloud Platform Integration:

1. **ThingSpeak Account:** A ThingSpeak account was created to store and visualize the sensor data.
2. **API Keys:** API keys were generated to authenticate the Arduino device with ThingSpeak.
3. **Data Transmission:** The Arduino code was modified to send the sensor data to ThingSpeak using the API keys.

ALGORITHM

□ **Sensor Connections:**

- **Pulse Sensor:** Connect the output pin to A0, VCC, and GND of the Arduino.
- **LM35 Temperature Sensor:** Connect the output pin to A1, VCC, and GND of the Arduino.

□ **LCD Connections:**

- Connect the LCD's pins 1, 3, 5, and 16 to GND.
- Connect pins 2 and 15 to VCC.
- Connect pins 4, 6, 11, 12, 13, and 14 to Arduino's digital pins 12, 11, 5, 4, 3, and 2, respectively.

□ **ESP8266 Connections:**

- **Voltage Divider:** Connect a 2.2kΩ resistor between the 5V supply and the RX pin of the ESP8266. Connect a 1kΩ resistor between the RX pin and GND.
- **Serial Communication:** Connect the RX pin of the ESP8266 to Arduino's pin 10 and the TX pin of the ESP8266 to Arduino's pin 9.

□ **LED Connection:**

- Connect an LED to digital pin 7 of the Arduino through a 220Ω resistor.

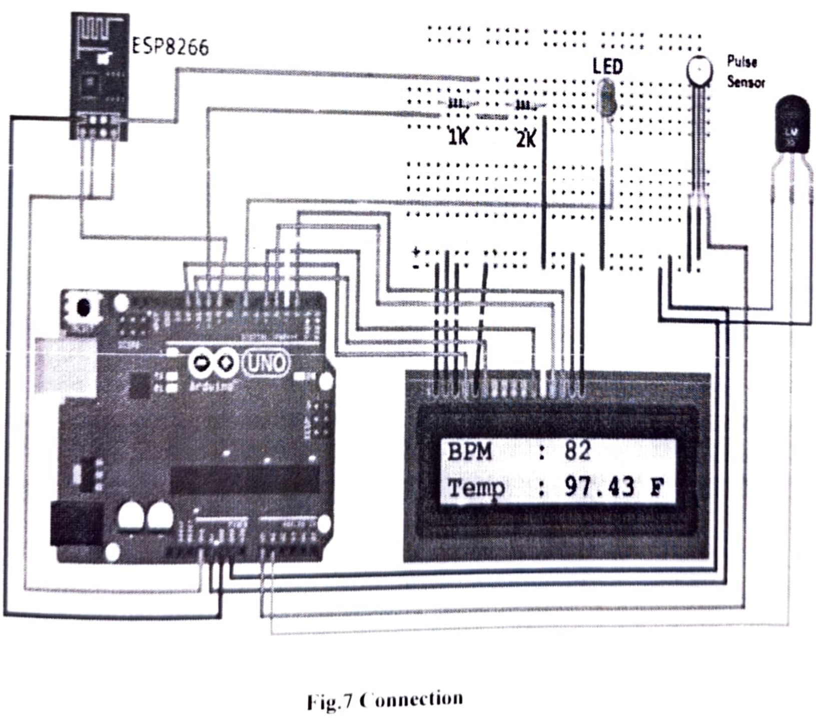

Fig.7 Connection

CHAPTER 5: MODELLING AND QUICK DESIGN

The modelling phase for the Arduino-powered health tracker involved designing a system that integrates multiple sensors and components to function seamlessly. This included:

Setting the ThingSpeak: ThingSpeak serves as a valuable platform for Internet of Things (IoT) projects. It allows users to monitor collected data and potentially control connected systems remotely. This functionality is achieved through "Channels" and associated web pages provided by ThingSpeak.

To get started, you'll need to create an account by visiting the ThingSpeak website (https://thingspeak.com). Once you've created a ThingSpeak account, establish a new channel and configure it according to your specific needs.

Next, generate API keys. These keys will be crucial for programming modifications and data input.

With the circuit assembled, upload the code to your Arduino Uno. Upon opening the serial monitor, the device will automatically connect to Wi-Fi and initiate the setup process.

Finally, access the "Channels" section on ThingSpeak to observe the live data stream from your IoT-based patient health monitoring system.

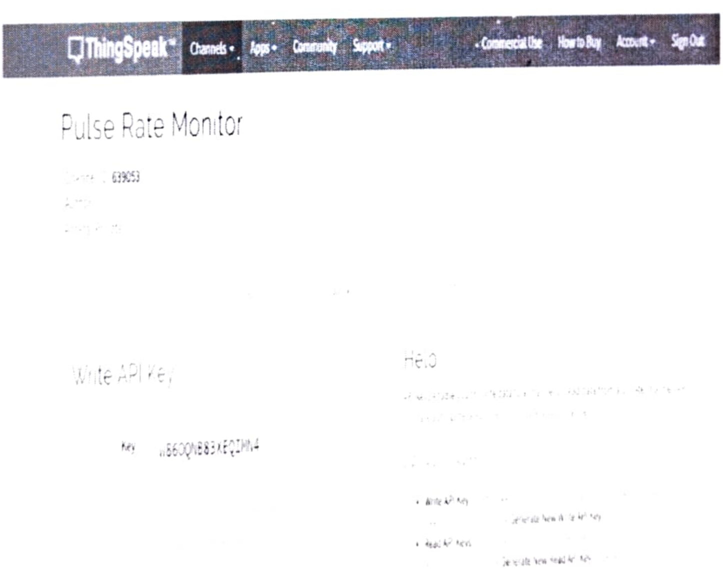

Fig.8 Setup ThingSpeak

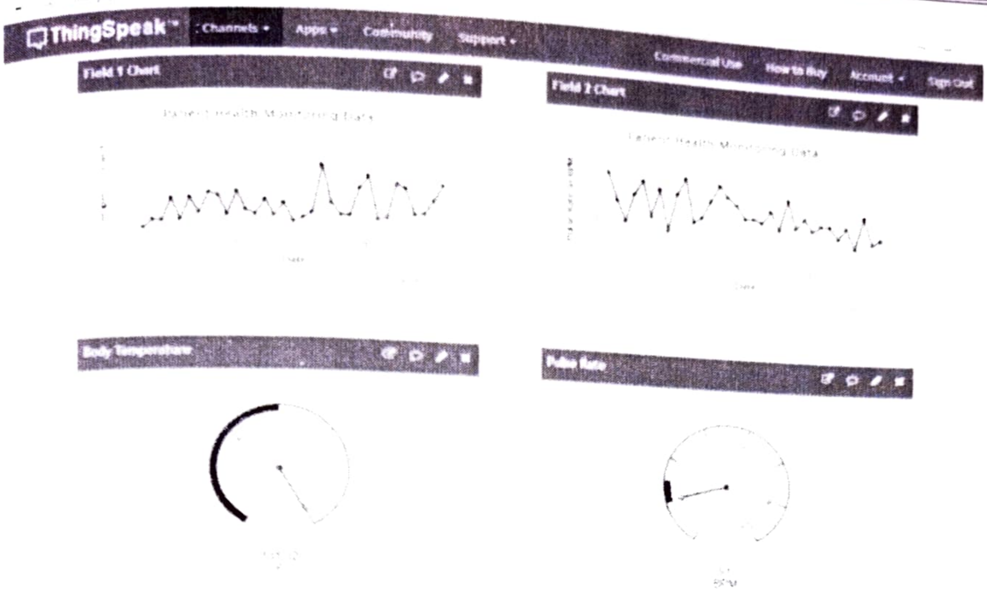

Fig.9 Result on ThingSpeak

Design Process:

System Diagram: To illustrate how the Arduino board and peripherals interact, a conceptual block diagram was created which is shown in Figure10. This stage made sure that communication protocols and pin assignments were clear .

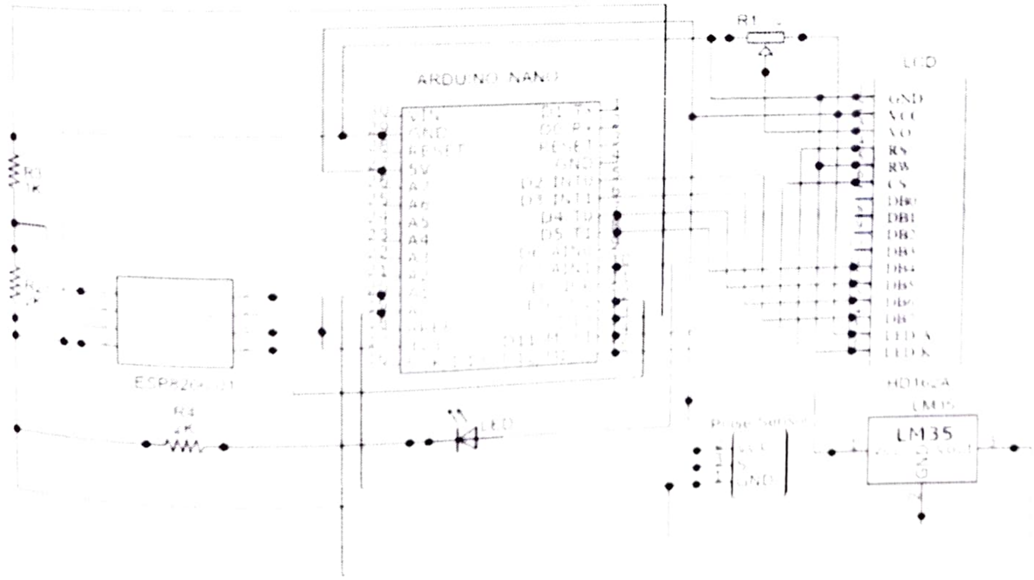

Fig.10 Design Process

CHAPTER 6: RESULTS ANALYSIS

The Arduino-based health monitoring system successfully demonstrated its ability to collect and transmit real-time health data, including heart rate, SpO2 levels, and temperature. The data was accurately captured by the sensors and transmitted to the ThingSpeak cloud platform.

Key Observations:

- **Sensor Accuracy:** The sensors used in the system exhibited satisfactory accuracy in measuring vital parameters.
- **Data Transmission:** The ESP8266 module effectively transmitted data to the ThingSpeak platform, ensuring reliable communication.
- **Cloud-Based Monitoring**: The ThingSpeak platform provided a user-friendly interface for visualizing and analyzing the collected data.
- **Real-Time Monitoring:** The system enabled real-time monitoring of health parameters, facilitating timely intervention.

Potential Improvements:

- **Enhanced Data Security:** Implement robust security measures to protect sensitive health data.
- **Battery Optimization:** Explore low-power techniques to extend the device's battery life.
- **User Interface:** Develop a more intuitive user interface for the device, possibly incorporating a display or mobile app.
- **Advanced Analytics:** Utilize advanced data analysis techniques to extract meaningful insights from the collected data.
- **Integration with Healthcare Systems:** Integrate the system with existing healthcare systems to enable remote patient monitoring and telemedicine.

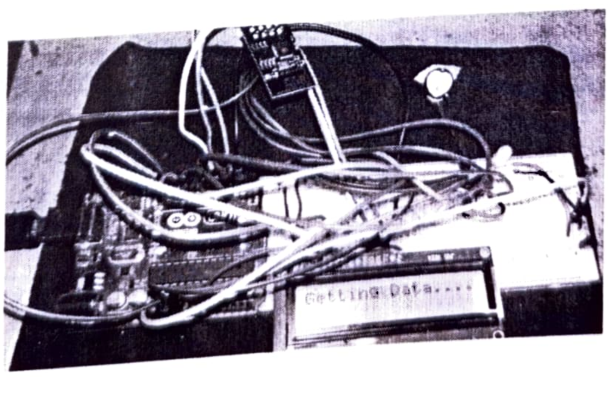

Fig. 11 Prototype Design

CHAPTER 7: CONCLUSION AND FUTURE SCOPE

This project successfully demonstrated the development of a low-cost, portable health monitoring system using Arduino. The system effectively collected and transmitted real-time health data, including heart rate, SpO2 levels, and temperature, to a cloud-based platform. The integration of the ESP8266 module enabled seamless wireless communication, allowing remote monitoring and analysis of the data. The user-friendly interface provided by ThingSpeak facilitated easy visualization and interpretation of health trends.

Future research and development efforts can further enhance the capabilities of this health monitoring system. Potential areas of improvement include:

Advanced Sensor Integration: Incorporating additional sensors, such as blood pressure monitors or electrocardiogram (ECG) devices, to provide a more comprehensive health assessment.

Enhanced Data Analytics: Implementing advanced data analysis techniques, such as machine learning, to identify patterns and predict potential health issues.

Improved User Interface: Developing a more intuitive and user-friendly interface, possibly through a mobile app or a dedicated web portal.

Enhanced Security: Strengthening security measures to protect sensitive health data from unauthorized access.

Integration with Healthcare Systems: Integrating the system with existing healthcare infrastructure to facilitate remote patient monitoring and telemedicine.

Power Optimization: Exploring energy-efficient techniques to extend the device's battery life.

REFERENCES

1. **Al-Anbagi et al., "IoT-Based Patient Health Monitoring System for Chronic Diseases," Sensors (MDPI), vol. 19, no. 12, p. 2845, Jun. 2019, doi: 10.3390/s19122845.** This paper details an IoT-based system using Arduino Uno for monitoring chronic illnesses. It discusses sensor selection, data transmission, and potential applications.
2. **M. A. Razak et al., "Wireless Sensor Network-Based Patient Health Monitoring System Using Arduino Uno," International Journal of Ubiquitous Computing and Applications (IJUCA), vol. 8, no. 4, pp. 301-310, 2017.** This research explores a patient health monitoring system with Arduino Uno using a wireless sensor network for data transmission. It focuses on hardware design and communication protocols.
3. **Arduino Project Hub: IoT based health monitoring system** (https://www.youtube.com/watch?v=padbdnkylIiA) This website provides a detailed tutorial on building an IoT-based health monitoring system with Arduino Uno. It includes component selection, code examples, and connection guides.
4. **Hackster.io: Build an IoT-based Patient Health Monitoring System** (https://www.hackster.io/bhanu11/an-iot-based-health-monitoring-system-ihms-9ff6d3) This platform offers a project guide for constructing an IoT patient health monitoring system. It emphasizes practical aspects like sensor integration and data visualization.
5. **National Institutes of Health (NIH): Telehealth Technologies** (https://www.ncbi.nlm.nih.gov/books/NBK459384/) This website from a reputable government source provides an overview of telehealth technologies, including remote patient monitoring systems. It offers insights into the benefits and applications of such systems in healthcare.

ANNEXURE

SELF-EVALUATION OF THE PROJECT

Month	Start date – End date (DD/MM/YY) - (DD/MM/YY)	Progress of Project
Month- 1	05/08/24 - 31/08/24	Topic Selection and Project planning
Month- 2	01/09/24 - 30/09/24	Hardware selection and Prototype Development
Month- 3	01/10/24 - 31/10/24	Data Analysis and Results
Month- 4	01/11/24 - 18/11/24	Documentation of the Project

19/11/2024

Dr.Aftab Ahmed Ansari
Assistant Professor
Center for IOT
MITS, Gwalior