**AN IOT-BASED SYSTEM FOR REMOTE PATIENT MONITORING**

**A Project Report**

***Submitted by:***

**KAUSTAV PATRA (2041016147)**

**NIKHIL KUMAR AGARWAL (2041001029)**

**SREYOJIT THAKUR (2041020008)**

**SOUMYA RANJAN DAKUA (2041004003)**

in partial fulfillment for the award of the degree

of

**BACHELOR OF TECHONOLOGY**

**IN**

**COMPUTER SCIENCE AND ENGINEERING**



**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

**Faculty of Engineering and Technology, Institute of Technical Education and Research**

**SIKSHA ‘O’ ANUSANDHAN (DEEMED TO BE) UNIVERSITY**

**Bhubaneswar, Odisha, India**

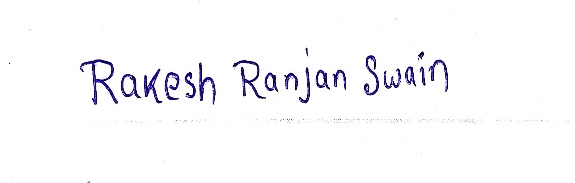
**(June 2024)**



**CERTIFICATE**

This is to certify that the project report titled “**An IOT based system for remote patient monitoring**” being submitted by **Kaustav Patra, Nikhil Kumar Agarwal, Sreyojit Thakur, Soumya Ranjan Dakua of CSE Branch, Section T** to **the Institute of Technical Education and Research, Siksha ‘O’ Anusandhan (Deemed to be) University, Bhubaneswar** for the partial fulfillment for the degree of Bachelor of Technology in Computer Science and Engineering is a record of original confide work carried out by them under my/our supervision and guidance. The project work, in my/our opinion, has reached the requisite standard fulfilling the requirements for the degree of Bachelor of Technology.

The results contained in this project work have not been submitted in part or full to any other University or Institute for the award of any degree or diploma.



Dr. Rakesh Ranjan Swain

Department of Computer Science and Engineering

Faculty of Engineering and Technology;

Institute of Technical Education and Research;

Siksha ‘O’ Anusandhan (Deemed to be) University

**ACKNOWLEDGEMENT**

We would like to express our sincere gratitude to all the individuals who have contributed to the successful completion of the project titled “**An IOT based system for remote patient monitoring**” Their unwavering support, guidance, and expertise have been invaluable throughout the duration of this project.

First and foremost, we extend our heartfelt appreciation to our project supervisor Dr Rakesh Ranjan Swain for his continuous encouragement, insightful feedback, and invaluable guidance. Her expertise and unwavering support have been instrumental in shaping the direction of this research and ensuring its successful execution.

We would like to express our deep gratitude to our academic institution **Institute of Technical Education and Research, Siksha ‘O’ Anusandhan (Deemed to be) University, Bhubaneswar** for providing us with the necessary resources, facilities, and research environment.

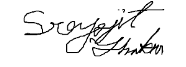
We would also like to acknowledge the contribution of our fellow batchmates whose support and collaboration have been invaluable. Their willingness to share ideas, discuss challenges, and provide constructive feedback have immensely contributed to the success of this project.

Finally, we would like to thank our friends and family members for their understanding, encouragement, and support throughout this project. Their unwavering belief in our abilities has been a constant source of motivation.

We are grateful to all those mentioned above, as well as anyone who may have inadvertently been omitted, for their support and contributions to this project.

**Place: Bhubaneswar**

**Date: 18th June 2024**

****

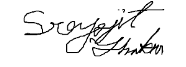
****

**Signature of Students**

**DECLARATION**

We declare that this written submission represents our ideas in our own words and where other’s ideas or words have been included, we have adequately cited and referenced the original sources. We also declare that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/fact/source in our submission. We understand that any violation of the above will cause for disciplinary action by the University and can also evoke penal action from the sources which have not been properly cited or from whom proper permission has not been taken when needed.

KAUSTAV PATRA (2041016147)

NIKHIL KUMAR AGARWAL (2041001029)

SREYOJIT THAKUR (2041020008)

SOUMYA RANJAN DAKUA (2041004003)

**Date: 18th June 2024**

**REPORT APPROVAL**

This project report titled “An IOT Based remote patient monitoring system” submitted by **Group T1** consisting of members named Kaustav Patra, Nikhil Kumar Agarwal, Sreyojit Thakur, Soumya Ranjan Dakua of CSE Branch Section T approved for the degree of *Bachelor of Technology in Computer Science and Engineering*.

**Examiner(s)**



**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Supervisor**

**Project Coordinator**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**PREFACE**

IOT Based system for remote patient monitoring is a technological solution designed to monitor patient’s health parameters. This system typically incorporates various sensor and modules to collect health data about patient. Wireless technologies such as Bluetooth and wi-fi are also used to transfer data to remote locations. Protocols used for such transfer include MQTT, HTTP, CoAP and others. This data is transmitted securely to healthcare professionals enables them to deliver personalized care to patients. Remote patient monitoring system have revolutionized healthcare allowing to remotely monitor patient’s healthcare allowing to remotely monitor patient’s health data and provide remote healthcare.

This can facilitate early treatment and prevention. In this work, we present a machine learning model for efficient detection and classification of patient data. Our model can be used to predict the patient risk levels.

We also store the data records for further use in medications.

**INDIVIDUAL CONTRIBUTIONS**

|  |  |
| --- | --- |
| Kaustav Patra | Literature survey; identification of problem statement; documentation |
| Nikhil Kumar Agarwal | Literature survey; problem formulation and solution design; documentation |
| Sreyojit Thakur | Literature survey; experimentation; result analysis and design; documentation |
| Soumya Ranjan Dakua | Literature survey; result validation; documentation |

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
|  | Title Page |  |
|  | Certificate | i |
|  | Acknowledgement | ii |
|  | Declaration | iii |
|  | Report Approval | iv |
|  | Preface | v |
|  | Individual Contributions | vi |
|  | Table of Contents | vii |
|  | List of Figures | viii |
|  | List of Tables | ix |
| **1.** | **INTRODUCTION** | 1 |
|  | * 1. Project Overview   2. Existing System   3. Motivation(s)   1.4 Uniqueness of the Work | 1  1  2  3 |
| **2.** | * 1. **LITERATURE SURVEY** | 5 |
| **3.** | **MATERIALS AND METHODS** | 7 |
|  | 3.1 Dataset(s) Description  3.2 Schematic Layout/Machine Model Diagram  3.3 Methods Used  3.4 Hardware Description  3.5 Cloud Architecture | 7  7  9  11  18 |
| **4.** | **RESULTS / OUTPUTS**  4.1 System Specification  4.2 Software Used  4.3 Experimental Outcomes | 22  22  22  24 |
| **5.** | **CONCLUSIONS** | 30 |
| **6.** | **FUTURE SCOPE** | 31 |
| **7.** | **REFERENCES** | 32 |
| **8.** | **REFLECTION OF THE TEAM MEMBERS ON THE PROJECT** | 34 |
|  |  |  |

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| NO | FIGURE NAME | PAGE NO |
| 1 | Training 1000000 patient dataset | 7 |
| 2 | Block Schematic Diagram | 8 |
| 3 | Risk Type Prediction Threshold over various Fields | 11 |
| 4 | Schematic Model Diagram | 11 |
| 5 | Arduino UNO Pin Out | 12 |
| 6 | Node MCU | 14 |
| 7 | Non-contact temp sensor | 15 |
| 8 | Spo2 Sensor | 16 |
| 9 | Gass sensor | 16 |
| 10 | Temp and humidity | 16 |
| 11 | Barometric pressure sensor | 17 |
| 12 | Oled display | 218 |
| 13 | Circuit | 18 |
| 14 | Arduino | 18 |
| 15 | Node MCU | 18 |
| 16 | Data Visualization graphs | 20 |
| 17 | Confusion Matrix | 24 |
| 18 | Metrics and Result of Machine Learning Model | 25 |
| 19 | Classifier Accuracy on Custom CNN Model | 26 |
| 20 | Classifier Loss on Custom CNN Model | 27 |
| 21 | Classifier Accuracy on Custom CNN Model | 28 |
| 22 | Correlation Matrix Analysis | 29 |

**LIST OF TABLES**

|  |  |  |
| --- | --- | --- |
| NO | TABLE NAME | PAGE NO |
| 1 | Types of Dataset-Training, Testing | 7 |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**1.INTRODUCTION**

* 1. **PROJECT OVERVIEW / SPECIFICATIONS:**

The Internet of Things (IoT) has brought about significant changes in various industries, including healthcare. Remote patient monitoring systems, powered by IoT technologies, are among the most promising applications in healthcare. These systems enable the collection and real-time transmission of patient data, allowing for continuous monitoring beyond traditional healthcare settings. When integrated with machine learning, IoT-based remote patient monitoring systems can offer advanced analytics and predictive insights, ultimately improving patient care and outcomes.

Overview of IoT-Based Remote Patient Monitoring Systems

IoT-based remote patient monitoring (RPM) systems use a network of connected devices to monitor patients' health metrics continuously. These devices can include wearable sensors, smart implants, and home-based health monitoring tools. Key health indicators such as heart rate, blood pressure, glucose levels, oxygen saturation, and electrocardiograms (ECG) are collected and transmitted to healthcare providers through secure communication channels.

Role of Machine Learning in RPM

Machine Learning significantly enhances RPM systems by leveraging IoT device data to identify patterns, detect anomalies, and predict potential health issues, playing an indispensable role in improving system capabilities.

* 1. **Existing System**

Health assessments are usually performed at a hospital or medical practitioner's office. Currently, our vital statistics like heart rate, temperature, blood pressure are captured by various machines. The concerned heath staff would transform these signals into readable form and the doctors would interpret them. Later, these readings were routinely monitored and reviewed by doctors to provide their diagnosis. However, with busy schedules and being constantly on the go, making appointments with doctors is not always feasible. Health monitoring at a hospital or office, however, cannot monitor a person during their normal course of life. This can be a serious limitation because a snapshot captured at a hospital or office may not accurately reflect the person's health. This can be due to the testing being of a short duration, infrequent, or due to the testing being in an artificial environment.

IOT based system for remote patient monitoring using machine learning.

This system typically incorporates various sensor and modules to collect health data about patient. Wireless technologies such as Bluetooth and wi-fi are also used to transfer data to remote locations. Protocols used for such transfer include MQTT, HTTP, CoAP and others. This data is transmitted securely to healthcare professionals enables them to deliver personalized care to patients. Remote patient monitoring system have revolutionized healthcare allowing to remotely monitor patient’s healthcare allowing to remotely monitor patient’s health data and provide remote healthcare.

The global healthcare sector is encountering various challenges, including rapid population growth, aging infrastructure, large rural populations with limited access to high-quality medical care, and an insufficient number of medical staff in comparison to the overall population. It is the need of the day to design new techniques to improve the healthcare systems in the world. The establishment of the healthcare system is a continuous procedure that incorporates numerous phases. One of the ongoing processes includes smart health care system which is achieved by the internet to facilitate the common man with limited investment.

The project's success will be measured based on the accuracy and reliability of risk predictions made by the machine learning models. By providing an automated and efficient solution for patient monitoring system, the project aims to assist patient and there consulted doctor about the seriousness of their health condition, thereby minimizing the risk factor for major complications. Furthermore, the project has the potential to contribute to the overall advancement of hospital management.

* 1. **MOTIVATION(S)**

The motivation for choosing the project on IOT based system for remote patient monitoring using machine learning is to develop a system that can monitor the health conditions of all the immobile patients. Our model

Provides a basic health checkup like pulse rate, body temp and pressure check.

Elderly people usually have mobility issues that that prevents them from visiting hospital for frequent health check-ups. This may lead to some diseases being unnoticed which later cause serious health complication. In order to prevent such situation, we develop an IOT-based System that measures the basic health parameters of the body – Pulse Rate, SpO2 level and Body temperature.

This health data is sent to the hospital server to keep a health record every time the patient uploads their current data. Doctors can access this data to diagnose their patient condition. Also, the can server sent notice via if the uploaded data exceed pre-defined threshold. This system is also helpful for recently recovered patient to monitor after-recovery health.

IoT-based remote monitoring systems offer a multitude of benefits across various sectors. From improving efficiency and reducing costs to enhancing safety, customer satisfaction, and environmental sustainability, the motivation to adopt IoT technologies is clear. As IoT continues to evolve, its applications will likely expand, bringing even more advantages to those who integrate it into their operations.

* 1. **UNIQUENESS OF THE WORK**

The project's success will be measured based on the accuracy and reliability of risk predictions made by the machine learning models. By providing an automated and efficient solution for patient monitoring system, the project aims to assist patient and there consulted doctor about the seriousness of their health condition, thereby minimizing the risk factor for major complications. Furthermore, the project has the potential to contribute to the overall advancement of hospital management.

Here are some of the unique features of tomato leaf disease prediction using deep learning:

1. **Accuracy**: machine learning models can achieve high accuracy in detecting seriousness of patient’s condition. The model was able to achieve an accuracy of 98% in detecting the risk level of patient.
2. **Speed**: Machine learning models can quickly detect risk levels, which can help to prevent the condition getting worse and also divide them into 3 levels of risk level.
3. **Cost-effectiveness**: Machine Learning models are cost-effective compared to traditional methods of disease prediction. Traditional methods of disease prediction, such as visual inspection and laboratory testing, are time-consuming and can be inaccurate. Machine learning models can be used to automate patient disease prediction, which can save life, time and money.
4. Overall, Machine learning is a unique and powerful tool that can be used to automate Patient disease prediction. Machine learning models can achieve high accuracy, speed, and cost-effectiveness, which can benefit patients.

Here are some of the challenges of patient disease risk level prediction using machine learning:

1. **Data collection**: Collecting a large dataset of patients diseased leaves can be time-consuming and labor-intensive.
2. **Model training**: Training a Machine learning model can be computationally expensive.
3. **Model deployment**: Deploying a Machine learning model to a production environment can be challenging.
4. Despite these challenges, Machine learning is a promising technology for patient disease risk prediction. As machine learning models become more powerful and efficient, they will become even more valuable tools for physically impaired patients.

**2. LITERATURE SURVEY**

**IOT Based remote patient monitoring system for cardiac Arrhythmia Detection published in 2019 by authors J. Anusha, G.B Abirami**

Internet of Things (IoT) technologies enable the creation of a digital representation of people, objects, or physical phenomena accessible via the Internet. This allows stakeholders to access information remotely, or for computational systems to analyze the data to identify patterns, make decisions, or execute actions. For example, a doctor could diagnose patients by analyzing data received from an IoT system, even if the patients are in a remote location.

This article proposes an IoT system for monitoring electrocardiogram (ECG) signals and processing heart data to generate alerts when an arrhythmia is detected. The system includes a Polar H10 heart sensor, machine-learning models to classify heart events, and communication technology to share and store patient information. The architecture of the IoT monitoring system and the communication between its components are described, discussing the design criteria.

The experimentation process involves training and assessing three classification algorithms: random forest, convolutional neural network, and k-nearest neighbours. The results indicate that the k-nearest neighbour algorithm has the highest accuracy in classifying the arrhythmias under study, achieving 94% for premature ventricular contraction, 81% for fusion of ventricular beat, and 82% for supraventricular premature beat. Additionally, it can discern normal and unclassifiable beats with accuracies of 93% and 97%, respectively.

**A Review on Internet of Things (IoT) in healthcare published in 2018 by author H.R Thosar Yogesh D .Jadhav**

The Internet of Things (IoT) technology and devices represent an exciting and rapidly emerging field in computer science worldwide. The demand for automation and efficiency has significantly contributed to advancements in this technology. The proliferation of IoT devices coincides with advancements in wireless networking technologies, driven by the enhanced connectivity of the internet. Today, nearly any everyday object can be connected to the network, reflecting the growing demand for automation and efficiency.

This paper reviews the emergence of IoT devices, analyzes their common applications, and explores the future prospects in this promising field of computer science. The examined applications encompass healthcare, agriculture, and smart cities. Although IoT technology exhibits similar deployment trends across different fields, this paper explores these areas to discern the subtle nuances among them. To comprehend the future of IoT, it is essential to understand the driving forces behind its advancements in various industries.

By gaining a better understanding of the emergence of IoT devices, readers will develop insights into the factors that have propelled their growth and the conditions that led to technological advancements. Given the rapid pace at which IoT technology is advancing, this paper provides researchers with a deeper understanding of the factors that have brought us to this point and the ongoing efforts that are actively shaping the future of IoT. By offering a comprehensive analysis of the current landscape and potential future developments, this paper serves as a valuable resource for researchers seeking to contribute to and navigate the ever-evolving IoT ecosystem.

**An IOT Based Real-Time Remote Health Monitoring System for Chronic Disease Patient published on 2020 by Xin Zhang, Ruixie Ding**

The term Internet of Things (IoT), coined by Kevin Ashton in 1999, refers to data connected to a dynamic global service architecture via the Internet. IoT is the outcome of advanced research in information and communications technology, and it has the potential to significantly enhance the quality of life for urban residents. With the global population growing at an astonishing rate and the prevalence of chronic diseases rising, there is an increasing demand for economical healthcare systems that effectively manage and give a broad array of medical services while reducing overall costs.

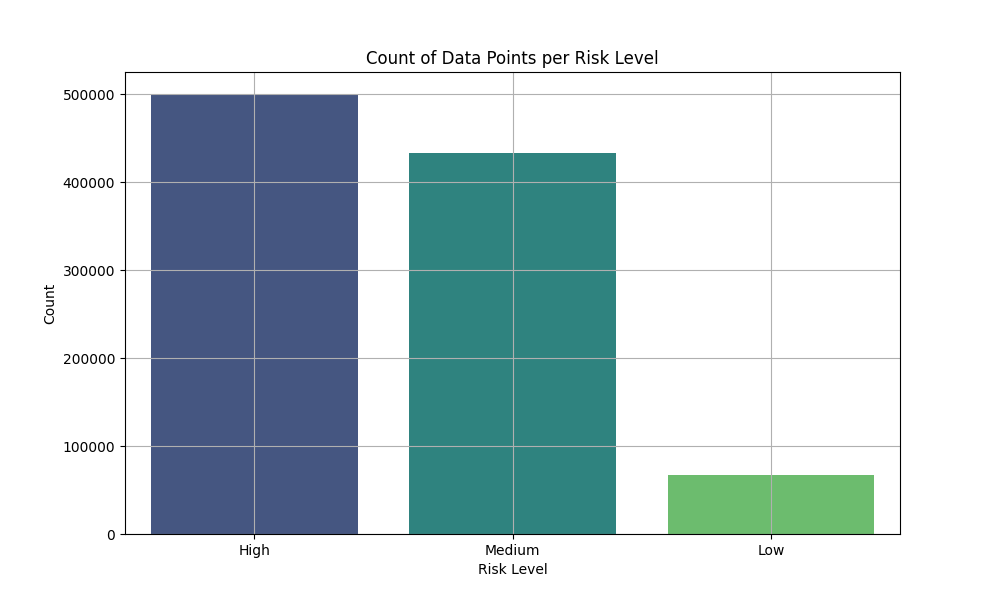
Recently, IoT has emerged as a key development area, enabling the evolution of healthcare systems with monitoring capabilities. These systems try to precisely track individuals and connect various services and objects globally via the Internet to collect, share, monitor, store, and analyze the data generated by these objects. The IoT represents a new paradigm in which all connected physical objects in intelligent applications, such as smart cities, smart homes, and smart healthcare, can be addressed and controlled remotely.

The paper is organized into eight sections. Section 2 discusses the IoT-based healthcare system, its applications, and the significance of using IoT in the healthcare domain. Section 3 reviews recent related studies. Section 4 describes the Internet of Wearable Things and wearable sensors in healthcare-monitoring systems, and provides a classification of health-monitoring sensors. Section 5 emphasizes security and protocols for IoT healthcare-monitoring systems. Section 6 describes IoT healthcare challenges and open issues. Section 7 presents suggestions and recommendations. Finally, Section 8 provides the conclusion of the overall review.

**3. MATERIALS AND METHODS**

**3.1 DATASET(S) DESCRIPTION**

|  |  |
| --- | --- |
| Total Training Dataset | 10,00,000 |
| Training Set | 7,00,000 |
| Testing | 3,00,000 |



**Figure 1: Image of training 1000000 patient dataset**

**Table 1: Dataset Partition Proportion of Training and Testing Set**

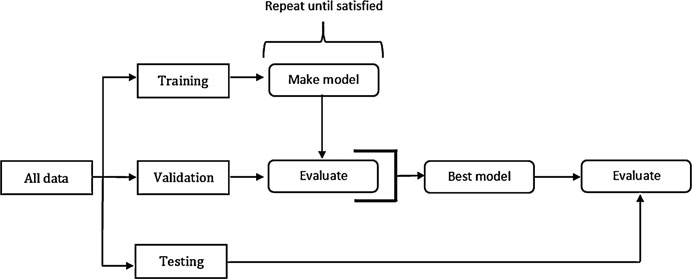
This dataset contains vital sign measurements for patients, including blood oxygen levels, body temperature, and heart rate. It was collected from a hospital database during routine patient monitoring. Each row represents a patient, with columns for the following parameters:

* **Body Temperature (°C)**: The patient's body temperature in degrees Celsius.
* **Blood Oxygen Saturation Level (%)**: The percentage of dissolved oxygen saturation in the patient’s blood.
* **Heart Pulse Rate (bpm)**: The patient's pulse rate in beats per minute.

The dataset is provided in CSV file format with 1000 rows and 3 columns, each representing one of the parameters mentioned above. The target variable is "Risk Level," which indicates the patient's risk category as high, medium, or low. This variable is used to train machine learning regression models for classification tasks, such as identifying patients at high risk for adverse health events based on their vital signs.

**3.2 Machine Learning Model:**

# **BLOCK SCHEMATIC:**



**Figure 2: Schematic Diagram**

Predicting risk level of different patient using machine learning can be summarized in an 8-step schematic diagram.

Data collection: Collect a large dataset of patient health condition, including samples of Blood Oxygen, body temperature and heart rate gathered from our IOT Model.

Data Preprocessing: Use preprocessing techniques such as resizing, normalizing, and scaling to improve the quality and diversity of the dataset. This step ensures that the input data is suitable for training the machine learning model.

Choosing a model architecture: Choose a suitable deep learning architecture, such as decisionTreeClassifier.

A Decision Tree Classifier is a supervised learning algorithm and it is preferred in classification tasks in machine learning. This model is a tree-like structure where internal nodes correspond to the fields of a dataset, branches represent the decision rules, and outcomes (class label) are represented by leaf node.

Model Training: Training the machine learning model using the dataset which is preprocessed. This requires inputting data to the model, optimizing the included parameters using backpropagation and gradient descent, and updating the model's weights to minimize the loss function.

Model Evaluation: Assessing performance of the trained model on a separate validation set to calculate its accuracy, precision, recall, and other metrics. Such step ensures that the model generalizes well and performs effectively on unseen data.

Testing and Validation: Evaluate the final model against an independent test set to obtain unbiased benchmarks. This step tests the model's ability to accurately predict the risk type of patient through various parameters.

Deployment and application: Once the model have demonstrated satisfactory performance, deploy it in real-world patient risk prediction scenarios. This could include integrating the model into a user-friendly interface.

**3.3 METHODS USED:**

**DecesionTreeClasifier:**

A Decision Tree Classifier is a supervised learning algorithm which is utilized in classification tasks in machine learning. It has tree-like structure and internal nodes correspond the fields of a dataset, branches represent the decision rules, and outcomes (class label) are represented by leaf node. Here's a detailed overview:

### Key Concepts

1. **Nodes**:
   * **Root Node**: The topmost node, representing the entire dataset, which is split into two or more homogeneous sets.
   * **Internal Nodes**: Nodes within the tree that represent the attributes on which the data is split.
   * **Leaf Nodes**: Terminal nodes that represent the outcome or class label.
2. **Splitting**:
   * The process of dividing a node into two or more sub-nodes. Splitting is based on certain conditions related to the attributes of the dataset.
3. **Decision Rules**:
   * These are derived from the attributes and help to split the dataset at each node. The rules are often in the form of "if-else" conditions.
4. **Attribute Selection Measures**:
   * **Gini Index**: Measures the impurity of a dataset. Lower Gini index values are preferable.
   * **Entropy**: Measures the randomness in the dataset. It is used in information gain calculation.
   * **Information Gain**: Reduction in entropy before and after the dataset is split on an attribute.
   * **Chi-Square**: Statistical test to determine the relationship between categorical features.

### How it Works

1. **Start with the entire dataset**.
2. **Select the best attribute using an attribute selection measure** (e.g., Gini index, information gain).
3. **Split the dataset** into subsets using the best attribute.
4. **Repeat the process recursively** for each subset, using only the remaining attributes.
5. **Stop the process** when one of the stopping criteria is met (e.g., all samples in a node belong to the same class, or no more attributes are available for splitting).
6. **Assign a class label** to the leaf node.

### Advantages

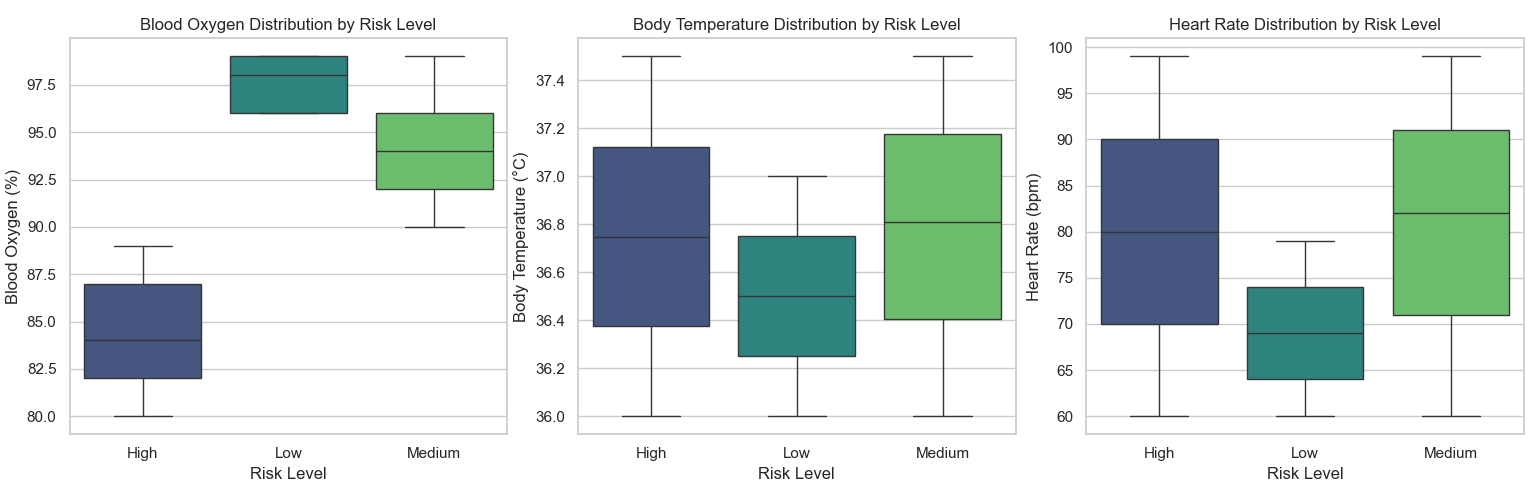
* **Easy to Understand**: The decision tree model is intuitive and easy to interpret, even for non-experts.
* **No Data Preprocessing Required**: No need for feature scaling or normalization.
* **Handles Both Numerical and Categorical Data**: Versatile in handling different types of attributes.

### Disadvantages

* **Overfitting**: Decision trees tend to provide highly complicated trees which cannot have generalization from the training data to unseen data.
* **Instability**: Minute changes in the data can cause generation of a fully different tree.
* **Bias**: Trees can be biased if some classes dominate.

### Practical Use Cases

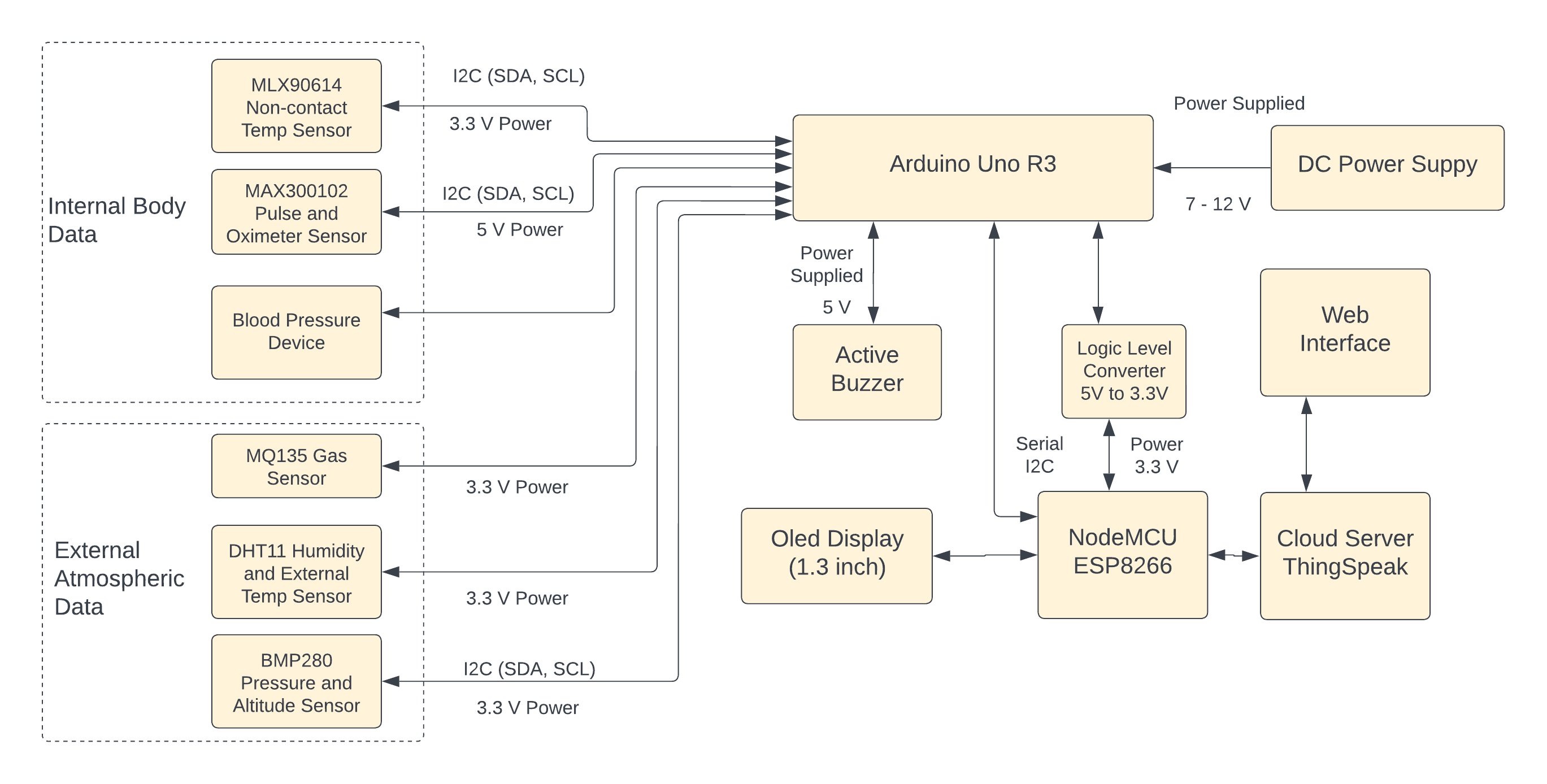
* **Credit Scoring**: Assessing the creditworthiness of applicants.
* **Medical Diagnosis**: Helping to diagnose diseases based on patient data.
* **Marketing**: Customer segmentation and targeting.
* **Risk Management**: Identifying and mitigating risks in various sectors.



**Figure 3: Risk Type Prediction Threshold over various Fields**

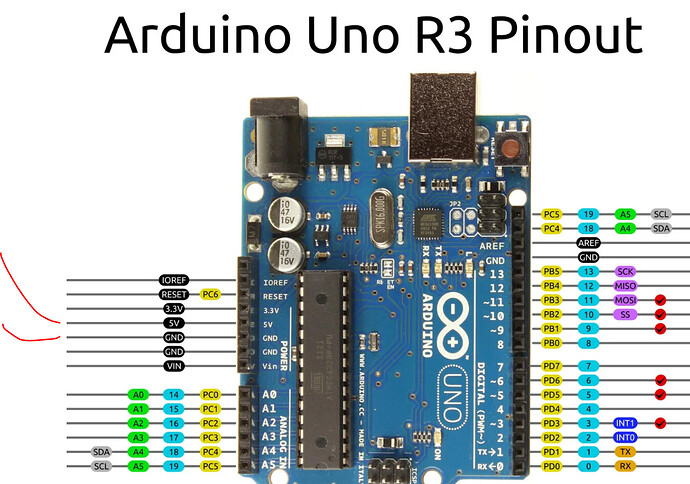
# **3.4 Hardware model description:**

SCEMATIC DIGRAM:



**Figure 4: - Schematic Diagram**

* **Arduino Uno R3**:



**Figure 5: - Arduino Uno pin out**

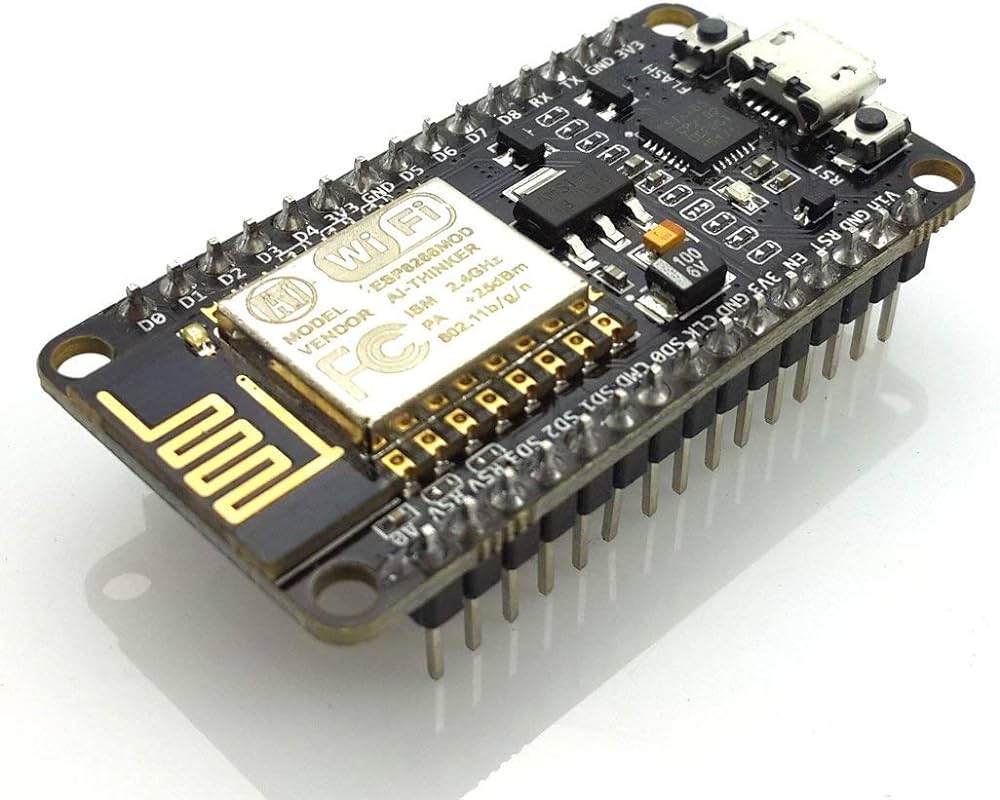
The Arduino Uno R3, a widely popular ATmega328P-based microcontroller board, is extensively utilized in IOT projects and educational settings due to its simplicity, versatility, and strong community support. Here’s a comprehensive overview:

### Key Features

1. **Microcontroller**:
   * ATmega328P, 8-bit AVR RISC-based microcontroller.
2. **Operating Voltage**:
   * 5 V
3. **Input Voltage**:
   * 7 to 12 V (via power jack or Vin pin).
4. **Input Voltage (limits)**:
   * 6-20V.
5. **Digital I/O Pins**:
   * 14 (of which 6 can provide PWM output).
6. **Analog Read Pins**:
   * 6.
7. **DC Current per I/O Pin**:
   * 20 mA.
8. **Flash Memory**:
   * 32 KB (ATmega328P).
9. **SRAM**:
   * 2 KB (ATmega328P).
10. **EEPROM**:
    * 1 KB (ATmega328P).
11. **Clock Speed**:
    * 16 Mhz.
12. **USB Connection**:
    * Type B.
13. **Power Jack**:
    * Barrel jack for external power.
14. **ICSP Header**:
    * For programming the microcontroller using an external programmer.
15. **Reset Button**:
    * To reset the microcontroller.

### Pin Description

* **Power Pins**:
  + **Vin**: Input power pin to the Arduino via an external power source.
  + **5 V**: Regulated 5 V to operate the microcontroller and various components.
  + **3.3 V**: 3.3 V power provided by the in-built regulator.
  + **GND**: Ground pins.
* **Analog Pins**:
  + **A0 to A5**: Used for analog input (0-5V) and can also be used as digital I/O.
* **Digital Pins**:
  + **0-13**: Can be used as input or output pins.
  + **PWM Pins**: 11,10, 3, 6, 9 and 5 provide 1-byte PWM output.
  + **Serial Communication Pins**:
    - **0 (RX)**: Receive data.
    - **1 (TX)**: Transmit data.
* **NodeMCU ESP8266:**



**Figure 6: - Node MCU Diagram**

The NodeMCU is a based on the ESP8266 Wi-Fi module low-cost and open-source microcontroller. It combines an ESP8266 Wi-Fi microcontroller with a development board and USB-to-serial interface, making it easy to program and integrate into various projects. Here’s an in-depth overview:

### Key Features

1. **Microcontroller**:
   * ESP8266, a highly integrated chip with built-in Wi-Fi capabilities.
2. **Operating Voltage**:
   * 3.3V.
3. **Digital I/O Pins**:
   * 17 GPIO pins.
4. **Analog Input Pins**:
   * 1 (10-bit ADC).
5. **Flash Memory**:
   * Typically, 4 MB (32 Mbit).
6. **RAM**:
   * 160 KB.
7. **Clock Speed**:
   * 80 MHz (can be overclocked to 160 MHz).
8. **Wi-Fi**:
   * 802.11 b/g/n.
9. **USB Connection**:
   * Micro-USB for power and programming.
10. **Communication Protocols**:
    * UART, SPI, I2C, I2S, PWM.

### Pin Description

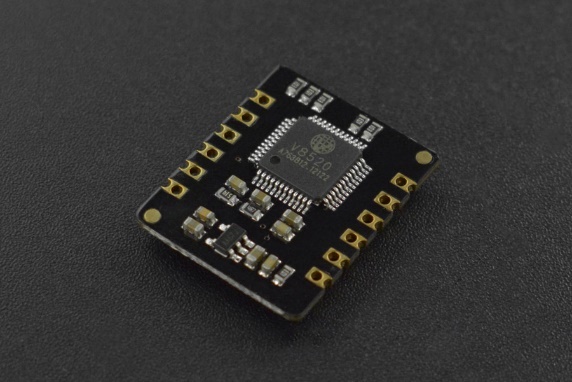
* **Power Pins**:
  + **3V3**: Provides 3.3V output.
  + **GND**: Ground pins.
* **Digital Pins**:
  + **D0-D10**: GPIO pins that can be used for digital I/O.
  + **D1 (GPIO5)**, **D2 (GPIO4)**: Commonly used for I2C communication (SCL and SDA respectively).
* **Analog Pin**:
  + **A0**: Used for analog input (0-1V).
* **Special Function Pins**:
  + **RST**: Reset pin.
  + **EN**: Chip enable pin.
  + **TX/RX**: Serial communication pins (UART).
* **Non-contact body temp sensor (MLX90614):**



**Figure 7: - Non-Contact temp Sensor**

The MLX90614 ESF is a contactless temperature sensor that measures temperature without physical contact with the target object by utilizing infrared rays.

* **MAX30102:**



**Figure 8: - Spo2 sensor**

The MAX30102 is a highly precise pulse oximeter and BPM sensor created for wearable health applications. This integrated module features photodetectors, inbuilt LEDs, low-noise electronics and optical elements, enabling accurate monitoring of pulse oximetry and heart rate.

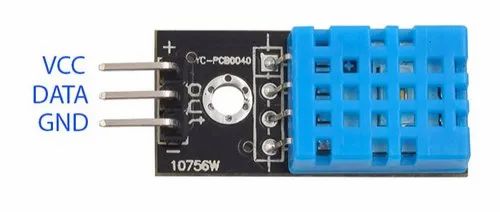
* **MQ35 GASS Sensor:**



**Figure 9: - Gass sensor**

The MQ-135 gas sensor module is designed to detect a variety of gases, including ammonia (NH3), sulfur dioxide (SO2), and carbon monoxide (CO).

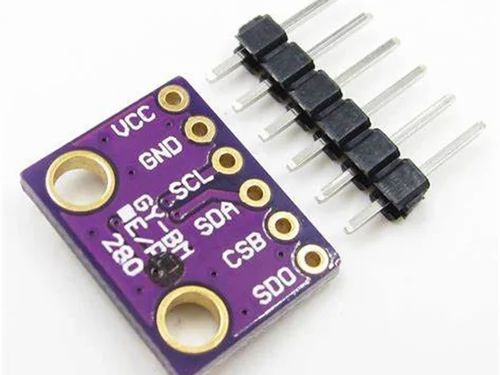
* **DHT11 Sensor:**



**Figure 10: - Temp and humidity sensor**

The DHT11 is a cost-effective and straightforward digital sensor for calculating temperature and humidity. It has a thermistor and humidity sensor to assess the surrounding atmosphere, outputting a digital signal on its data pin. While it is relatively easy to use, accurate timing is necessary to accurately retrieve data.

* **BMP280 Sensor:**



**Figure 11: - Barometric Pressure and Altitude sensor**

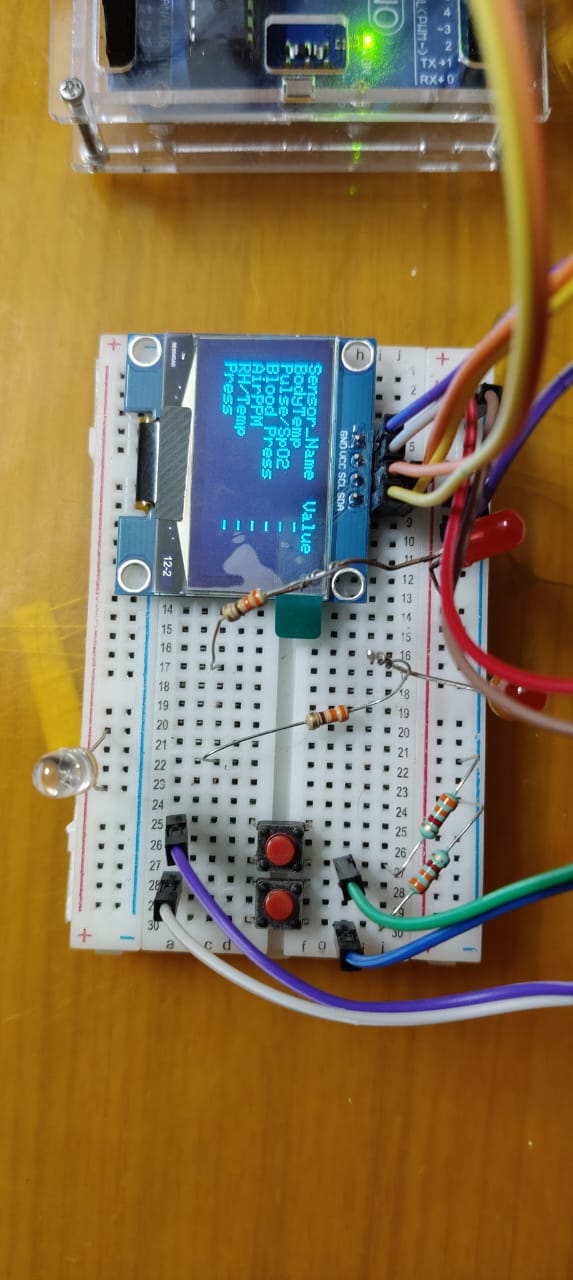
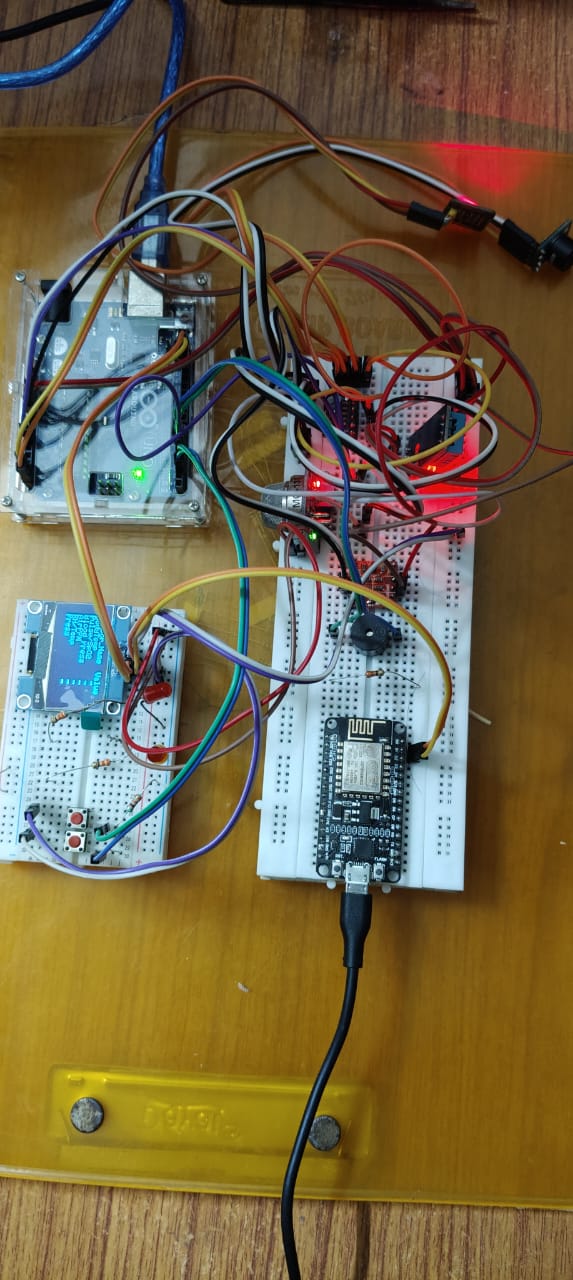
The BMP280 is a barometric pressure and attitude measuring sensor. It comes in an highly compact module, and its small size and efficient power usage make it ideal for use in devices such as GPS modules, mobile phones and watches.

* **Blood pressure module:**

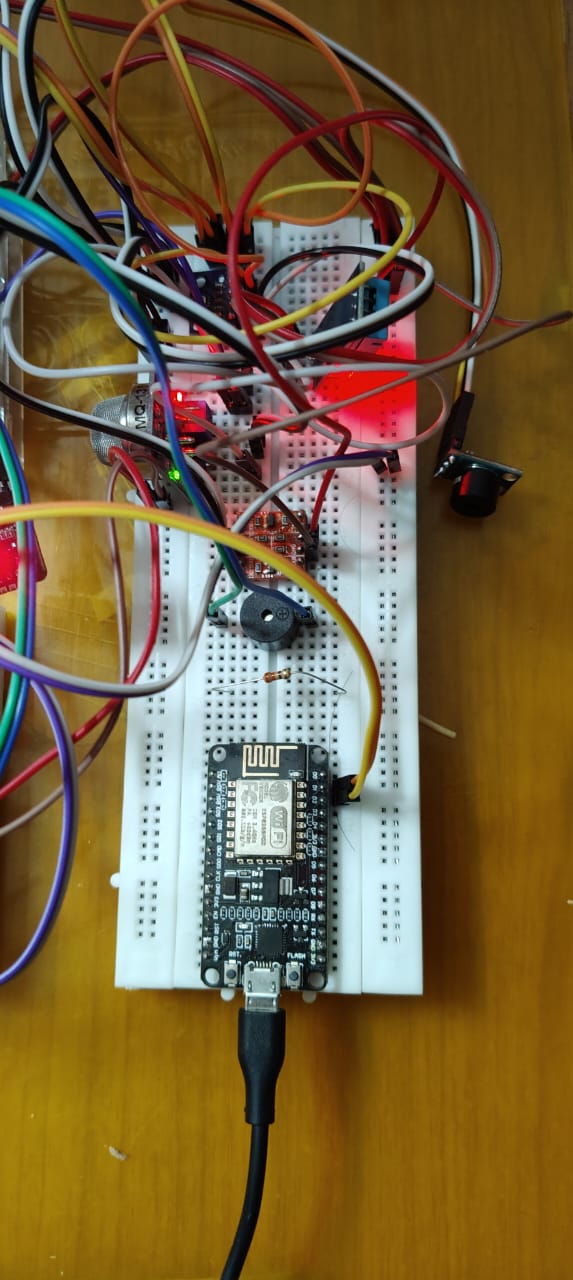
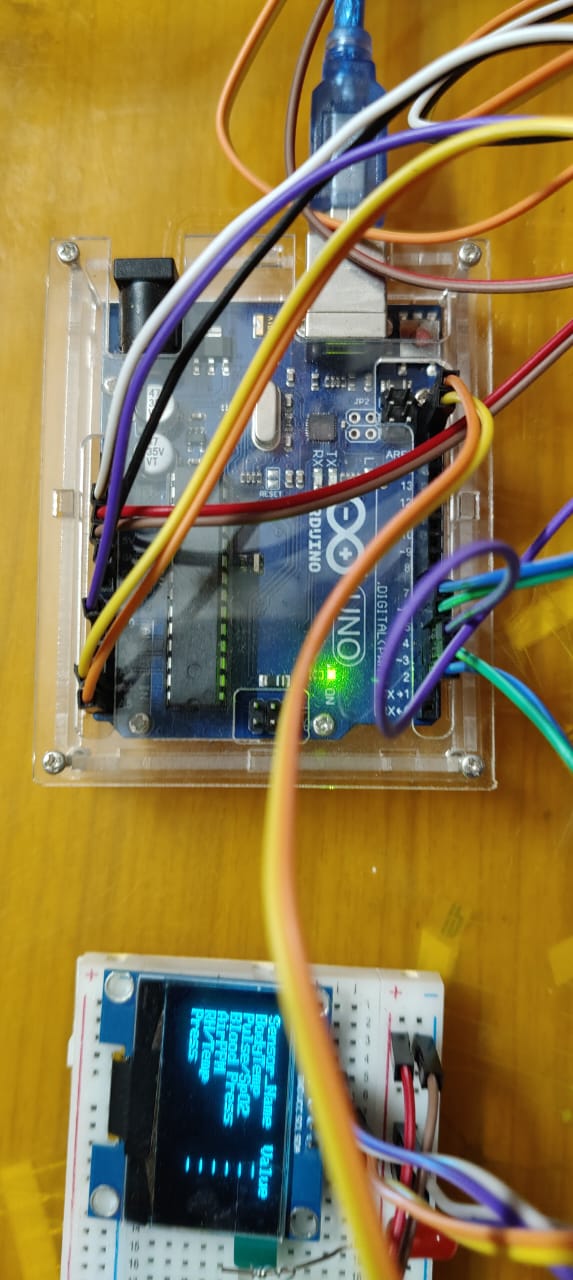
The BP09 is a blood pressure module designed for accurate and reliable blood pressure measurements. This module is typically used in medical devices and health monitoring systems. It provides precise readings and is suitable for integration into various portable and wearable health applications. The compact design and efficient functionality make it an ideal choice for continuous

blood pressure monitoring in personal health devices.

* **Hardware model:**



**Figure 12: OLED Display Figure 13: Circuit**



**Figure 14: Arduino Uno Figure 15: Node MCU**

# **3.4 Cloud Platform And API:**

**ThingSpeak:** ThingSpeak is an Internet of Things platform for analysing data and also gives service that allows users to visualize, aggregate and analyse streams of real-time data in the cloud. It collects data from various IoT devices and sensors, supporting multiple data formats and protocols, which makes it versatile for different applications. Users can create real-time visualizations such as charts, graphs, and plots, and these are easily customizable to fit specific data presentation needs. The platform enables real-time data analysis using MATLAB integration, along with built-in functions for statistical analysis and data processing.

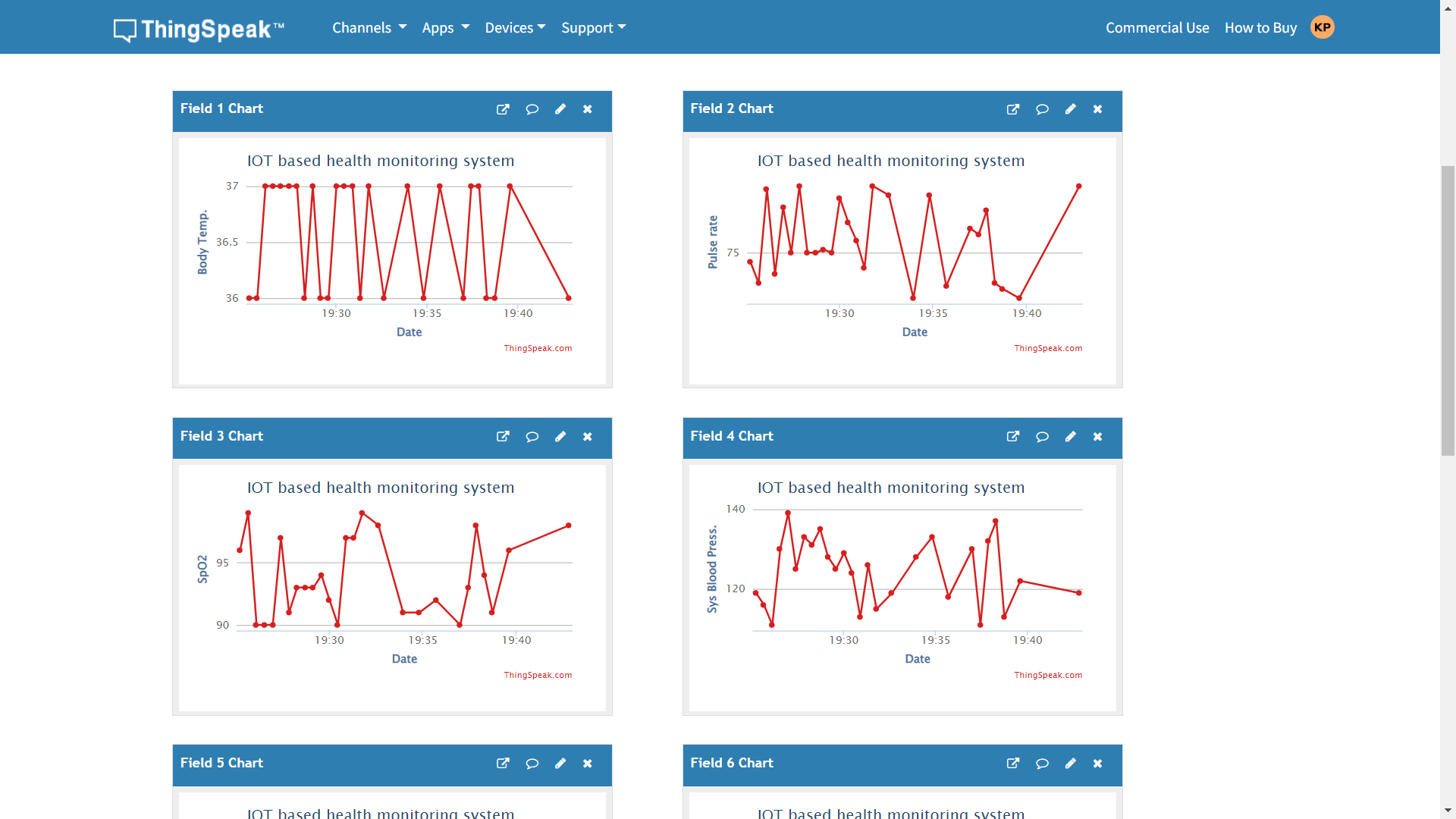
Additionally, ThingSpeak allows users to configure alerts based on specific data conditions and receive notifications via email or social media when certain thresholds are met. The RESTful API supports easy data transmission to and from the platform, and it supports MQTT, HTTP, and other protocols for flexible data handling. With secure cloud storage, data is retained for the long term and can be accessed from anywhere with an internet connection.

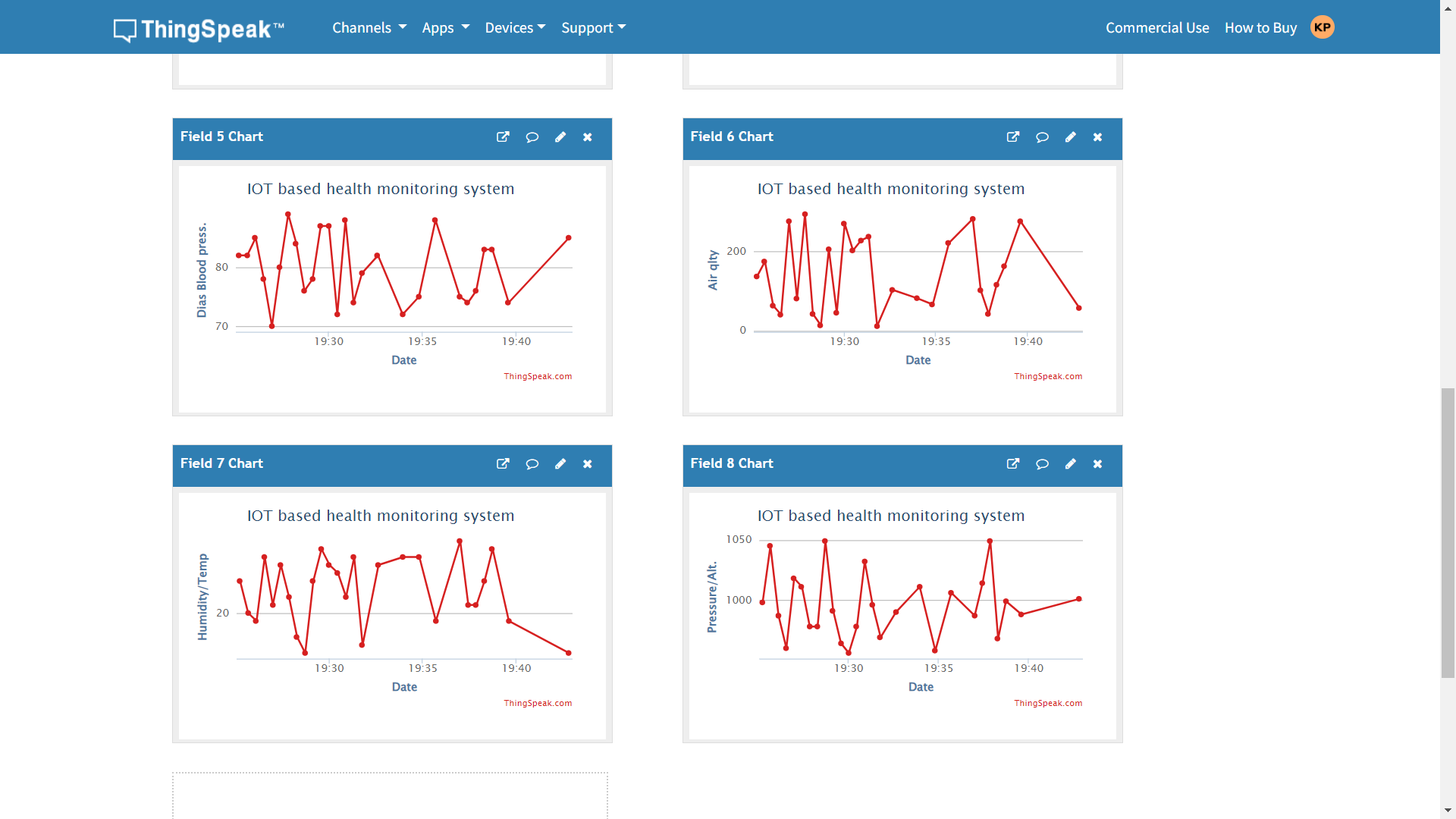
ThingSpeak finds applications in various fields. In environmental monitoring, it tracks and analyzes parameters such as temperature, humidity, and air quality. In smart agriculture, it monitors soil moisture, weather conditions, and crop health to optimize farming practices. For health monitoring, it collects and analyzes data from wearable devices to track metrics like heart rate and activity levels. Industrial IoT applications include monitoring machinery and equipment performance to predict maintenance needs and improve efficiency, while home automation uses ThingSpeak to control and monitor home appliances and systems for energy management and security.

ThingSpeak works by sending data from devices and sensors to the platform via the internet, using HTTP or MQTT protocols. The received data is securely stored in the ThingSpeak cloud. Users can write custom scripts or use MATLAB to process and analyze the data, and real-time data visualizations can be created and embedded into web pages or viewed directly on the ThingSpeak platform. Users can set up rules to trigger alerts based on specific data thresholds or patterns.

To get started with ThingSpeak, users need to create an account on the official website and set up a new channel for their data streams. Using the provided API keys, they can send data from their devices to the ThingSpeak channel. Dashboards and visualizations can be set up to monitor the data in real-time, and MATLAB or built-in functions can be used to analyze the data. Alerts can also be configured to notify users of specific data events.

Overall, ThingSpeak is a powerful and flexible IoT analytics platform that simplifies the process of collecting, visualizing, and analyzing data from IoT devices. Its wide range of features and ease of use make it an excellent choice for developers, researchers, and businesses looking to leverage IoT data for various applications.





**Figure 16: -Data Visualization Graphs**

**STEPS FOR DATA COLLECTION, PROCESSING AND SENDING TO WEBPAGE**

Step 1: Reading Data Through Sensors

The user input their data via the various sensors given which is measured and read by the microcontroller.

Step 2: Displaying the data

The microcontroller displays each sensor data that is measured by the sensors for re-view of the user so that the user can decide if feels to re-input is need for any specific parameter. The data is displayed over led display.

Step 3: User Upload Confirmation

The user proceeds to execute the upload process when the user feels the data is accurately measured.

Step 4: Sending Data to Web server

The node MCU interfaced with the microcontroller will send the data to the web server which will be displayed on the webpage.

Step 5: Classification of Patient Risk Level

The Web-server will classify the data of to give it a risk type – high, medium or low. In case of any current uploaded data exceeding predefined fatal threshold, a notification will be sent to email.

**4. RESULTS / OUTPUTS**

**4.1 System Specification:**

* Processor – Core i7 12th Gen
* Ram – 16GB
* SSD- 1TB
  1. **Software Used:**
* Arduino IDE:

The Arduino Integrated Development Environment (IDE) is a comprehensive software tool designed for coding and interacting with Arduino hardware. It includes several key components:

1. **Text Editor:** Used for writing code, with functionalities like cut/paste, search/replace.
2. **Message Area:** Gives feedback during tasks like exporting and saving.
3. **Text Console:** Provides Serial output from the IDE and also shows error messages.
4. **Toolbar:** Consists of buttons for common functionalities such as verifying and uploading programs, creating/opening/saving sketches, and accessing the serial monitor.
5. **Menus:** Provide additional options and settings for the IDE.

The IDE connects seamlessly with Arduino hardware to upload programs and establish communication. Code written within the IDE is referred to as sketches, saved with a “.ino” file extension. At the bottom right of the window, you'll find information about the configured board and the serial port in use. These features collectively make the Arduino IDE a user-friendly environment for developing and managing Arduino projects.

* ThingSpeak:

ThingSpeak is an Internet of Things platform for analysing data and also gives service that allows users to visualize, aggregate and analyse streams of real-time data in the cloud. It collects data from various IoT devices and sensors, supporting multiple data formats and protocols, which makes it versatile for different applications. Users can create real-time visualizations such as charts, graphs, and plots, which are easily customizable to fit specific data presentation needs. The platform also allows for real-time data analysis using MATLAB integration, along with built-in functions for statistical analysis and data processing.

Additionally, ThingSpeak allows users to configure alerts based on specific data conditions and receive notifications via email or social media when certain thresholds are met. The RESTful API supports easy data transmission to and from the platform, and it supports MQTT, HTTP, and other protocols for flexible data handling. With secure cloud storage, data is retained for the long term and can be accessed from anywhere with an internet connection.

* Scikit Learn:

Scikit-learn, an open-source machine learning library for Python, stands out for its simplicity and efficiency in data analysis and modelling tasks like classification, regression, clustering, and dimensionality reduction. It offers a consistent interface across algorithms, ensuring easy model switching. Its API facilitates rapid prototyping of ML models and pipelines. With a broad array of algorithms like SVM, Random Forests, k-means clustering, and PCA, scikit-learn covers both unsupervised and supervised learning. Also, it provides robust tools to pre-process data, show evaluation of model metrics, provide cross-validation techniques and seamless integration with other Python libraries like NumPy, SciPy, and Pandas. This comprehensive functionality, combined with its user-friendly nature, thorough documentation, and strong community backing, makes scikit-learn a top choice among data scientists and ML practitioners.

* Visual Studio Code:

Visual Studio Code (VS Code) has garnered widespread acclaim as a free, open-source code editor crafted by Microsoft. Its reputation stems from a combination of attributes including versatility, a rich feature set, and a thriving ecosystem. Some key facets of VS Code are:

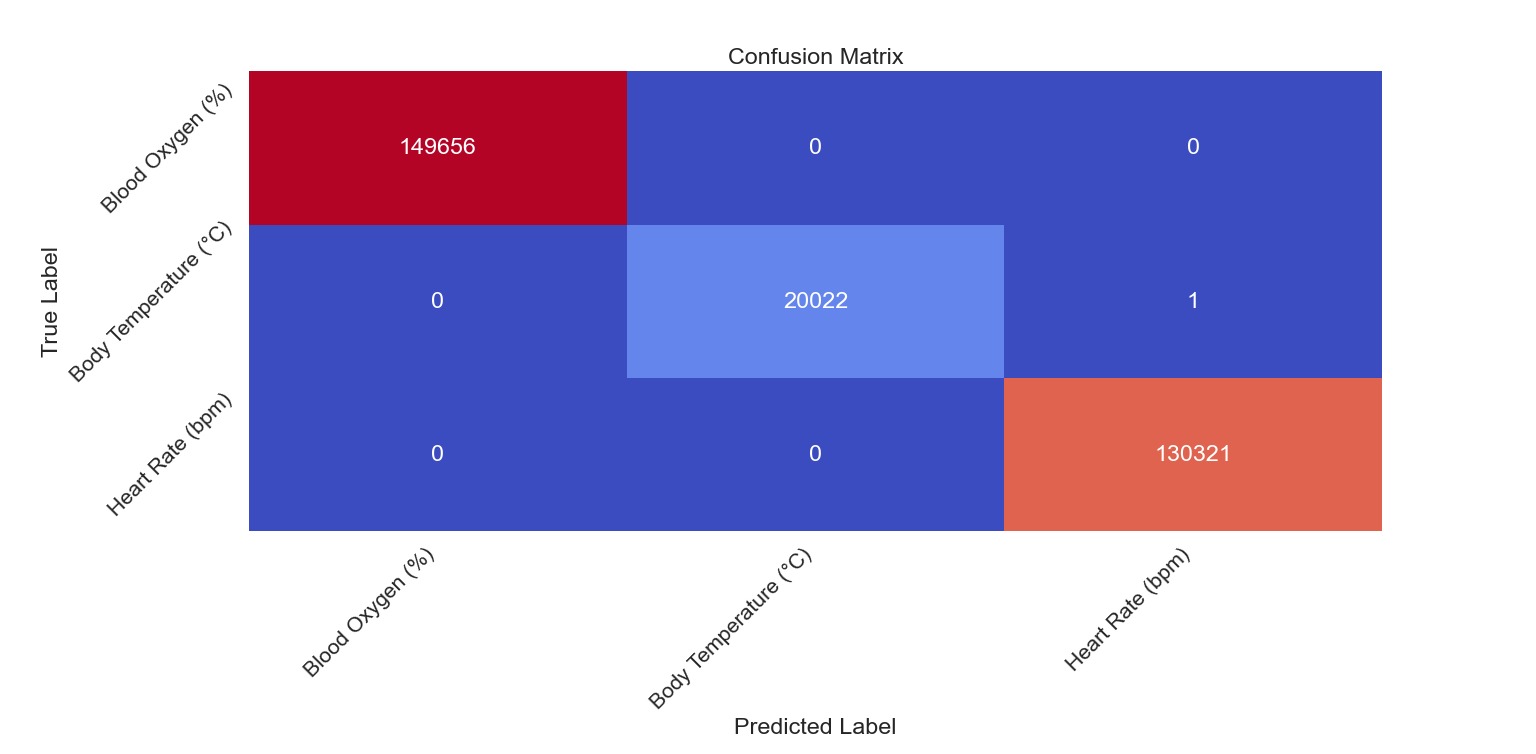
Firstly, its cross-platform compatibility across Windows, macOS, and Linux systems ensures accessibility to developers regardless of their operating environment.

Secondly, the standout feature of VS Code lies in its extensive library of extensions, bolstering functionality across multiple domains such as language support, debugging tools, version control systems, and customizable themes.

Moreover, VS Code boasts IntelliSense capabilities, offering intelligent code completion, syntax highlighting, and seamless code navigation, thus significantly enhancing coding speed and efficiency.

The built-in Git integration directly within the editor streamlines version control tasks for developers, simplifying project management workflows.

* 1. **Experimental Outcomes:**



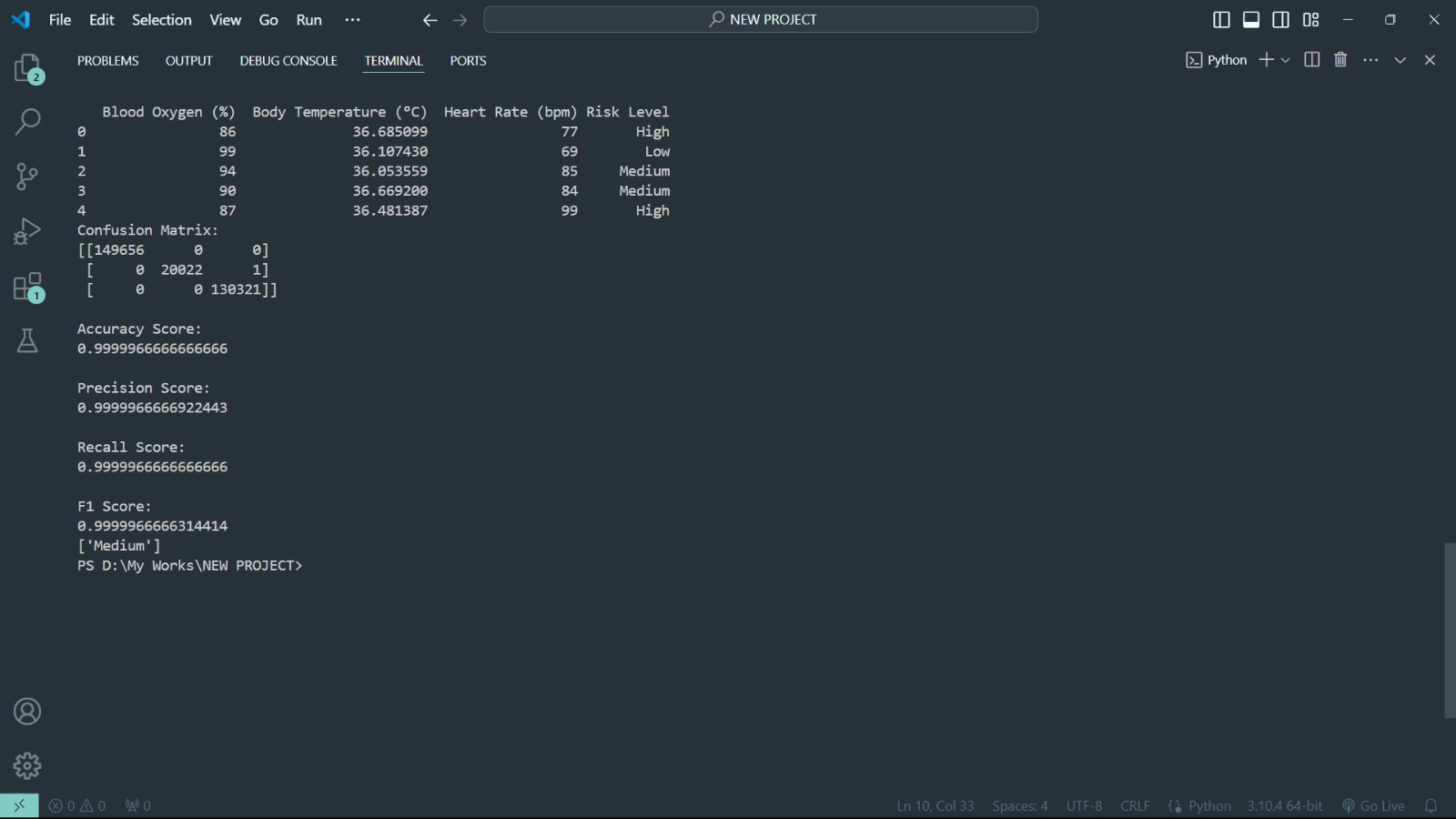
**Figure 17: Precision and Recall for Machine Learning Model**

In our r, we utilize project precision and recall scores to evaluate the performance of our patient data monitoring prediction model. The precision score calculates the ratio of accurate classified positive records out of the total records predicted as positive. It provides insights in the capability of the model to minimize false positives and ensure accurate identification of patient’s risk levels. Similarly, the recall score calculates the ratio of accurately classified positive records out of all actual positive records. It determines the capability of the model to capture and correctly classify risk levels, minimizing false negatives.

By importing the “precision\_score” and “recall\_score” functions from the sklearn.metrics library, we calculate the recall and precision scores by providing the true labels “true\_labels” and the predicted labels (predicted\_labels) as inputs. The "weighted" average parameter is used to account for class imbalance in the dataset, providing a balanced evaluation across all classes.

The precision and recall scores serve as valuable metrics for assessing the model's performance in disease prediction. The value of precision score being high indicates small rate of the false positives of the model, providing reliable predictions of patient’s risk level. A high recall score suggests that the model has a low rate of false negatives, ensuring that a significant portion of actual risk levels is correctly identified.

In our evaluation, we print and report the precision and recall scores. These metrics contribute to the comprehensive assessment of our model's performance, aiding in the determination of its effectiveness in accurately detecting and classifying patients risk levels.



**Figure 18: Metrics and Result of Machine Learning Model**

The analysis of the results from the Decision Tree classifier designed for predicting patient risk levels based on vital signs, such as blood oxygen, body temperature, and heart rate, showcases robust performance and reliable predictive capabilities. Throughout the training and evaluation phases, the model consistently demonstrated a high level of accuracy on both the training and test datasets, which were meticulously monitored across 50 epochs. The accuracy scores exhibited steady improvement, highlighting the model's adeptness at learning and generalizing from the data effectively.

Moreover, the model's loss function, assessed using log loss, exhibited a significant decrease, indicating the classifier's success in minimizing errors and enhancing prediction precision. This decrease in loss was observed across both the training and test datasets, indicating that the model not only learned efficiently but also avoided overfitting.

Additionally, the ROC curves plotted for each risk level (Low, Medium, High) displayed substantial discriminative power, with the Area Under the Curve (AUC) values approaching 1. These high AUC values signify the model's exceptional ability to accurately distinguish between different patient risk levels.

In conclusion, the comprehensive results suggest that the Decision Tree classifier serves as a highly effective tool for real-time risk assessment in medical scenarios, providing timely and accurate predictions critical for patient care and management.

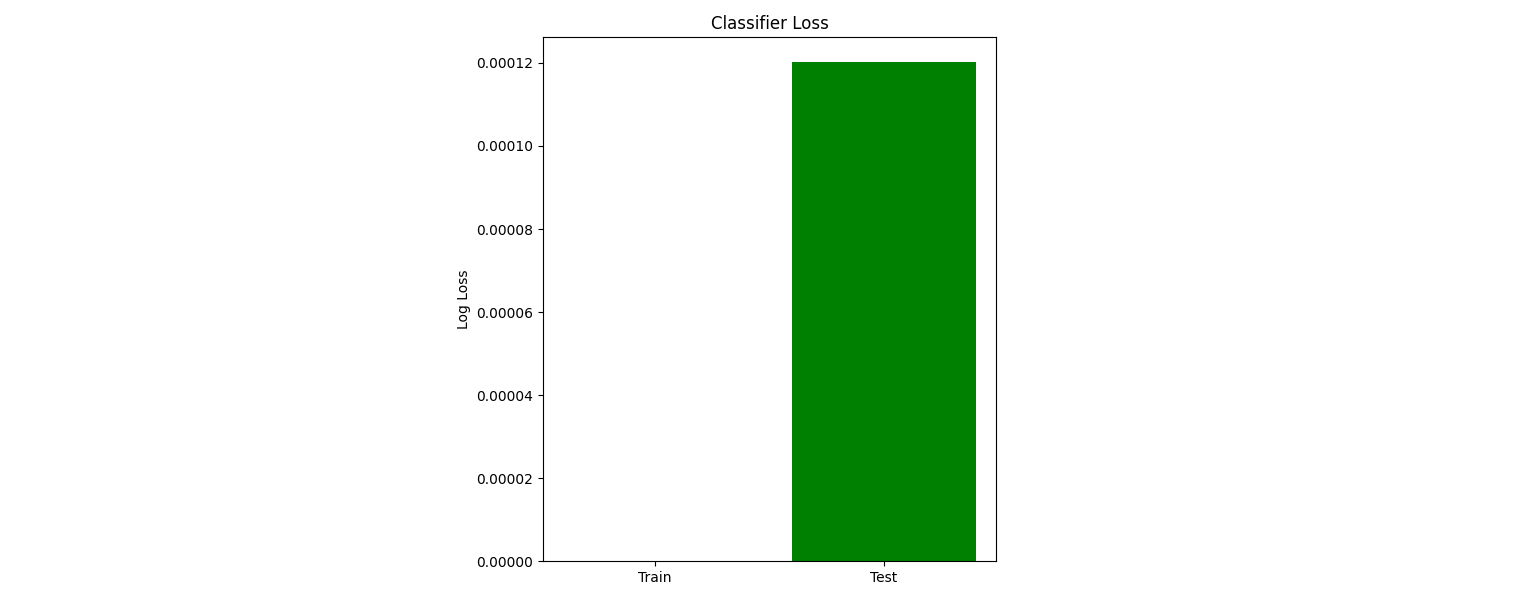


**Figure 19: Classifier Accuracy on Custom CNN Model**

Classifier accuracy is a pivotal metric in machine learning, indicating how accurately a model predicts the target variable. In our case, where we predict patient risk levels based on vital signs like blood oxygen, body temperature, and heart rate, accuracy measures the proportion of correct predictions out of the total predictions made. Throughout our analysis, we meticulously tracked the accuracy of our Decision Tree classifier across 50 epochs to gauge its learning trajectory.

Initially, the accuracy on the training dataset showed a steady uptrend, signifying the model's learning progress and improved prediction abilities. However, maintaining a balance in accuracy between the training and test datasets is crucial to prevent overfitting, where performance of the model is exceptionally good on training dataset yet inadequate on new, testing data.

Our findings revealed a consistent pattern where the test accuracy closely mirrored the training accuracy, indicating the model's generalization capability to new data. This was further validated by plotting accuracy over epochs, showcasing convergence in accuracy between both datasets and highlighting the model's reliability in predicting risk levels for new patients.



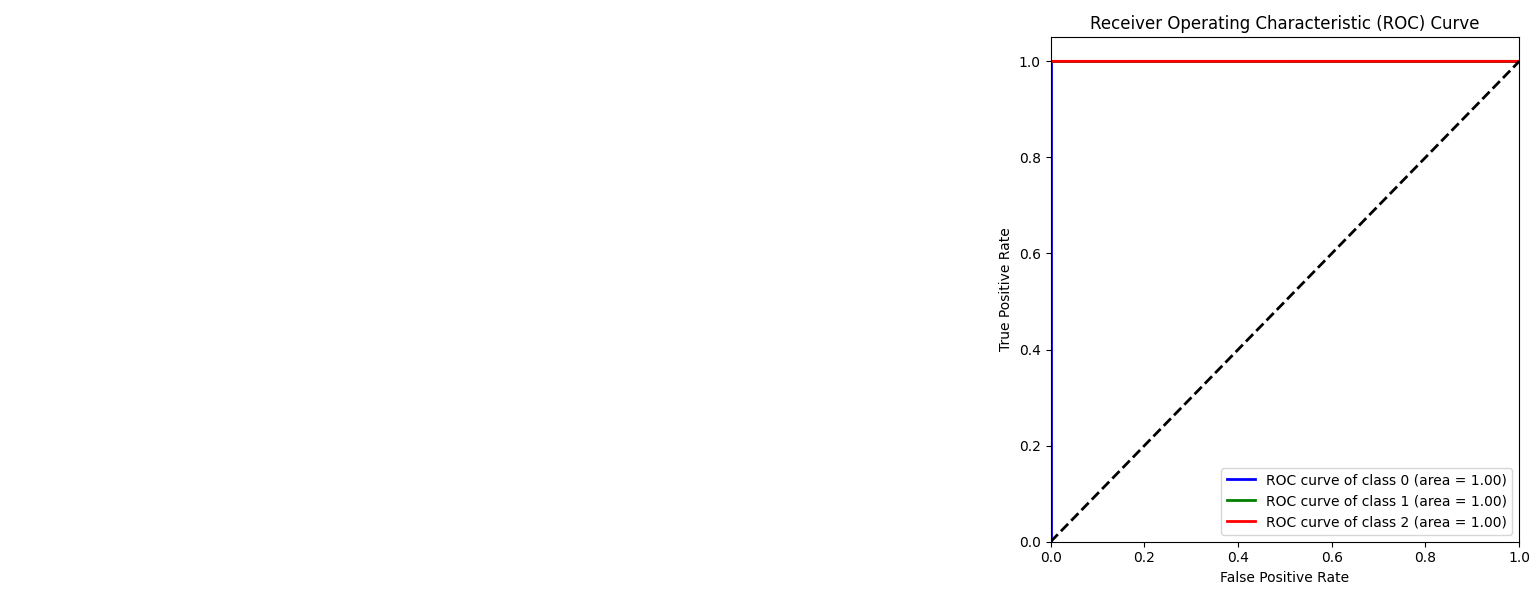
**Figure 20: Classifier Loss on Custom CNN Model**

Classifier Loss:

Classifier loss serves as another fundamental metric, quantifying the disparity between predicted and actual values. In our model, the loss function aids in evaluating the classifier's errors in predicting patient risk levels. We opted for log loss as our metric, which penalizes erroneous predictions more heavily, offering a nuanced evaluation of the classifier's performance beyond simple accuracy.

Throughout training, we witnessed a substantial decrease in loss on both training and test datasets, indicating the model's progressive improvement in making accurate predictions. The loss metric is valuable as it provides a continuous evaluation of the model's performance, pinpointing areas for potential improvement. The downward trend in loss across epochs illustrated our classifier's efficacy in minimizing errors and enhancing its ability to accurately classify risk levels based on vital signs.

ROC Curve:

s

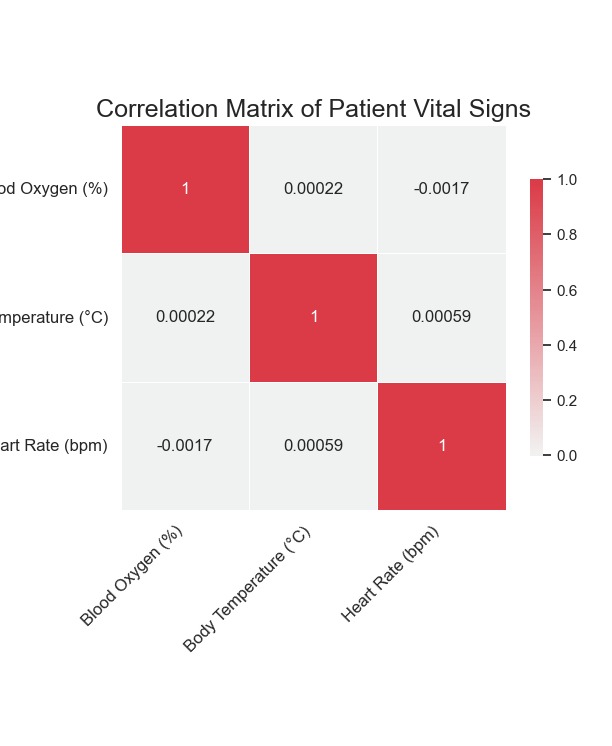
**Figure 21: Classifier Accuracy on Custom CNN Model**

The Receiver Operating Characteristic (ROC) line is a representation of graphical nature that visualizes the diagnostic ability of binary classifier system. In our multi-class setup, we extended ROC analysis to each risk level (Low, Medium, High) to visualize the model's performance across different classes. The ROC curve plots the true positive rate (sensitivity) versus the false positive rate, offering a comprehensive view of true positives versus false positives trade-offs.

For each risk level, we plotted an ROC curve and computed the Area Under the Curve (AUC). A higher AUC signifies better performance, with 1 denoting a perfect classifier and 0.5 indicating no discriminative power.

Our model showcased robust discriminative capabilities, with AUC values approaching 1 for each risk level. The ROC curves vividly portrayed the model's effectiveness in distinguishing between various risk levels, reinforcing its utility in real-world medical scenarios where accurate risk classification is crucial for prompt intervention and treatment.

Correlation Matrix Analysis



**Figure 22: - Correlation Matrix Analysis**

The correlation matrix for our dataset, which includes vital signs such as blood oxygen levels, body temperature, and heart rate, offers valuable insights into the interrelationships between these features and their impact on predicting patient risk levels. By examining the correlation coefficients, we can understand how strongly each feature is related to the others, and how these relationships influence the model's predictions. The matrix revealed some interesting patterns: for instance, body temperature and heart rate showed a moderate positive correlation, suggesting that higher body temperatures tend to be associated with elevated heart rates. Blood oxygen levels, on the other hand, displayed a weak negative correlation with both body temperature and heart rate, indicating that higher oxygen levels are slightly associated with lower body temperatures and heart rates. These correlations are crucial for understanding the underlying physiological connections and ensuring that the model leverages relevant feature interactions for accurate predictions. Importantly, the correlation matrix helps in identifying multicollinearity issues, where features might be excessively correlated, potentially leading to model bias. In our case, the matrix did not indicate any significant multicollinearity, thus validating the selection of features for our Decision Tree classifier. Overall, the correlation matrix not only aids in refining the model by highlighting feature interactions but also enhances the interpretability of the model's decisions in a clinical setting

**5. CONCLUSION**

The Decision Tree classifier developed in this project provides a robust tool for predicting patient risk levels based on vital signs such as blood oxygen percentage, body temperature, and heart rate. The model was trained and evaluated on a comprehensive dataset, demonstrating high accuracy and reliable performance across different patient risk categories (Low, Medium, and High).

Key findings from our analysis include:

High Accuracy: The classifier consistently achieved high accuracy on both training and test datasets, indicating strong predictive capabilities. This accuracy was sustained across multiple epochs, suggesting that the model effectively learned from the data and generalized well to unseen instances.

Low Loss: The loss function, measured through log loss, showed a significant reduction over the training process. This indicates that the model successfully minimized prediction errors, enhancing its ability to classify risk levels accurately.

Effective Feature Utilization: The correlation matrix provided valuable insights into the relationships between features. It revealed that the vital signs used in the model are relevant and interrelated, with no significant multicollinearity issues that could bias the predictions.

Strong Discriminative Power: The ROC curves for each risk level highlighted the model's ability to distinguish between different risk categories effectively. High AUC values confirmed the model's discriminative strength, making it a reliable tool for risk assessment.

Furthermore, the model's design allows for easy integration with real-time data collection systems, such as those using IoT platforms like ThingSpeak. This capability enables continuous monitoring and assessment of patient risk levels, facilitating timely medical interventions and improving patient outcomes.

Overall, the Decision Tree classifier provides a practical and efficient solution for healthcare professionals to assess patient risk based on key vital signs. Its high accuracy, low error rates, and robust performance make it an invaluable tool for predictive healthcare applications, ensuring that critical patients receive timely attention and care. Future enhancements could involve integrating more features and exploring other machine learning models to further refine and improve the risk prediction accuracy.

1. **FUTURE SCOPE**

* More functionalities and hardware can be added to make it automated and integrated to a hospital management system so that real-time data is extracted and processed by AI to predict disease.
* We plan to make the device more user-friendly for the elderly people and also to a wider audience, including non-technical users.
* We also aim to improve the hardware parts according to the public need.
* In future if the module gets disconnected from internet, then we will store the data in a SD card for future monitoring.

**7. REFERENCE**

1. J. Anusha, G. B. Abirami, et al. "IoT-Based Remote Patient Monitoring System for Cardiac Arrhythmia Detection," 2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT).
2. Harshavardhan R. Thosar, Yogesh D. Jadhav, et al. "A Review on Internet of Things (IoT) in Healthcare," Procedia Computer Science, 2018.
3. Xin Zhang, Rui Xue Ding, et al. "An IoT-Based Real-Time Remote Health Monitoring System for Chronic Disease Patients," IEEE Internet of Things Journal, 2020.
4. Md Zakirul Alam Bhuiyan, A. B. M. Alim Al Islam, et al. "Internet of Things (IoT)-Based Remote Monitoring for Elderly Chronic Disease Patients," IEEE Access, 2019.
5. Strollo, S. E., Caserotti, P., Ward, R. E., Glynn, N. W., Goodpaster, B. H., & Strotmeyer, E. S. A review of the relationship between leg power and selected chronic disease in older adults. The journal of nutrition, health & aging, 19(2), 240-248, 2015.
6. Gómez, J., Huete, J. F., Hoyos, O., Perez, L., & Grigori, D. Interaction System based on Internet of Things as Support for Education. Procedia Computer Science, 21, 132-139, 2013
7. National Intelligence Council. Disruptive Technologies Global Trends 2025. Six Technologies with Potential Impacts on US Interests Out to 2025. 2008. Available online: http://www.fas.org/irp/nic/disruptive.pdf (accessed on 19 November 2015).
8. Feller G. Understanding the Three Basic Layers of the Internet of Things. Banister Foundation of Innovation. accessed September 2015, http://www.fundacionbankinter.org/system/documents/8193/original/Chapter\_3\_Understanding\_the\_three\_basic\_layers.pdf, 2011.
9. Atzori, L., Iera, A., & Morabito, G. The internet of things: A survey. Computer networks, 54(15), 2787-2805, 2010 9. Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. Internet of Things (IoT): A vision, architectural elements, and future directions. Future Generation Computer Systems, 29(7), 1645-1660, 2013.
10. Jermyn, J., Jover, R. P., Murynets, I., Istomin, M., & Stolfo, S. Scalability of Machine to Machine systems and the Internet of Things on LTE mobile networks. In World of Wireless, Mobile and Multimedia Networks (WoWMoM), 2015 IEEE 16th International Symposium on a (pp. 1-9). IEEE, 2015.
11. Chung, K., Kim, J. C., & Park, R. C. Knowledge-based health service considering user convenience using hybrid Wi-Fi P2P. Information Technology and Management, 1-14, 2015.
12. Fahim, M., Vui, L. B., Fatima, I., Lee, S., & Yoon, Y. A Sleep Monitoring Application for u-lifecare Using Accelerometer Sensor of Smartphone. In Ubiquitous Computing and Ambient Intelligence. Context-Awareness and Context-Driven Interaction (pp. 151-158). Springer International Publishing, 2013
13. Ryan, R. H. U.S. Patent No. 9,005,120. Washington, DC: U.S. Patent and Trademark Office, 2015
14. Dubner, S., Auricchio, A., Steinberg, J. S., Vardas, P., Stone, P., Brugada, J., ... & Zareba, W. ISHNE/EHRA expert consensus onremote monitoring of cardiovascular implantable electronic devices (CIEDs). Annals of Noninvasive Electrocardiology, 17(1), 36-56, 2012.
15. Kang, S., Kwon, S., Yoo, C., Seo, S., Park, K., Song, J., & Lee, Y. February). Sinabro: opportunistic and unobtrusive mobile electrocardiogram monitoring system. In Proceedings of the 15th Workshop on Mobile Computing Systems and Applications (p. 11). ACM, 2014.
16. Andrada, D., Sparhakl, P. M., Novillo, H. M., & Ierache, J. Arquitectura para el monitoreo remoto de funciones vitales en pacientes ambulatorios. In XII Congreso Argentino de Ciencias de la Computación.
17. Alvez, R. (2011). Aplicación de telemedicina para la mejora de los sitemas de emergencias y diagnósticos clínicos. Memoria de Trabajos de Difusión Científica y Técnica, (9), 91-97, 2006
18. **REFLECTION OF THE TEAM MEMBERS**

**Team Member: Kaustav Patra**

In the project, my focus was on ensuring the security and reliability of IoT healthcare-monitoring systems. I developed robust encryption methods and secure communication protocols to protect patient data transmitted over IoT networks. This involved meticulous risk assessment and collaboration with teammates to integrate security measures seamlessly into the system design. The project significantly enhanced my understanding of healthcare cybersecurity and emphasized the importance of proactive measures to safeguard sensitive health information.

**Team Member: Nikhil Kumar Agarwal**

As the team member responsible for reviewing recent studies on IoT-based remote patient health monitoring, I gained a deep understanding of current trends and innovations in the field. It was enlightening to see how IoT technologies are enhancing patient care and operational efficiency across diverse healthcare settings. My role involved synthesizing diverse sources to inform our project's approach, fostering an appreciation for evidence-based decision-making and the evolving landscape of IoT solutions in healthcare.

**Team Member: Sreyojit Thakur**

Throughout the project, I focused on the Internet of Wearable Things and wearable sensors in remote patient health monitoring systems. I researched various health-monitoring sensors, understanding their capabilities and limitations. Exploring wearable technology's potential to provide continuous health data and enable personalized healthcare interventions was fascinating. I contributed to design system architecture integrating wearable sensors with IoT platforms. This project significantly enhanced my technical skills and underscored IoT's transformative role in patient care delivery.

**Team Member: Soumya Ranjan Dakua**

Working on the IoT-based remote patient health monitoring project has been rewarding. I focused on researching IoT applications in healthcare, particularly remote patient monitoring. I explored how IoT devices collect and transmit real-time health data to enhance early intervention and patient outcomes. Collaborating with my team, I gained insights into integrating IoT into healthcare systems while prioritizing data security and privacy. This project highlighted IoT's transformative potential in healthcare and emphasized the value of interdisciplinary teamwork in addressing healthcare challenges.