# Patient Reminder System

Vishakha Salve

This project is all about ESP8266 based Patient Health Monitoring System using MAX30100 Pulse Oximeter sensor, DS18B20 temperature sensor, and DHT22 Temperature & Humidity sensor. This system will monitor the parameters like room temperature, room Humidity, Heart Rate, Oxygen Saturation (Sp02) in blood, and body temperature of patients.

## Overview

Patient Health Monitoring on ESP8266 Healthcare technology is very popular in this pandemic situation because of coronavirus. Actually, health care technology is rapidly being revolutionized with the help of the Internet of Things (IoT). Monitoring the health status of a covid patient is a hard task because of our busy schedule and our daily work. Mostly, the elderly covid patients should be monitored periodically. So, I thought to make an innovative system in this lockdown to automate the task.

This device uses an ESP8266 webserver to track patient health using this monitoring system. Hence, patient health parameters such as body temperature, heart rate (BPM), blood oxygen levels (Sp02) as well as room temperature and humidity can be monitored from any device (like Smartphone, PC, Laptop, Smart TV,.) That support browsing capabilities. In this project, we will learn how to build an ESP8266 based Patient Health Monitoring System. To measure Heart Rate/Pulse (BPM) and Blood Oxygen Level (SpO2), we use the MAX30100 pulse oximeter sensor. Similarly, to measure body temperature, we use the DS18B20 temperature sensor. Meanwhile, the patient is inside the room. So we need to monitor room temperature and humidity level as well. We should keep them in a room with a certain temperature and humidity level to not feel uncomfortable. Hence, we use the DHT22 Temperature & Humidity sensor. A Smart Patient Health Monitoring System using IoT (Internet of Things) is a technology-driven solution designed to continuously collect, transmit, and analyze health-related data from patients. It provides real-time insights to healthcare professionals, patients, and caregivers to improve patient care, enhance early disease detection, and promote overall well-being. One important component of such a system is the alert system, which notifies relevant parties when certain health parameters or conditions require attention.

Here's an overview of the Patient Health Monitoring System using IoT:

1. **Data Collection:** The IoT system gathers data from various medical devices and sensors, such as wearable devices, blood pressure monitors, heart rate monitors, thermometers, and more. These devices collect data on vital signs, activity levels, and other relevant health metrics.

2. **Data Transmission:** Collected data is transmitted securely to a central server or cloud-based platform through wireless communication protocols such as Wi-Fi, Bluetooth, or cellular networks. The data is then processed and analyzed in real-time.

3. **Data Analysis:** Advanced algorithms and machine learning models are used to analyze the incoming data. The system can detect patterns, anomalies, and critical health events based on predefined thresholds and medical guidelines.

4. **Alert Generation:** When the system identifies a deviation from normal health parameters or detects a critical event (e.g., abnormal heart rate, sudden drop in oxygen levels, irregular blood sugar levels), it generates alerts. These alerts can be categorized based on severity: -

**Critical Alerts:** These are for life-threatening situations or immediate medical attention. Examples include a sudden cardiac arrest, severe hypoxia, or a diabetic coma. –
**High Priority Alerts:** These indicate a significant health concern that requires prompt attention but may not be immediately life-threatening. For example, an abnormal heart rhythm or high fever. **Low Priority Alerts:** These are for less urgent issues or trends in health data that healthcare providers should monitor over time. Examples include a gradual increase in blood pressure or a rising body temperature.
**5. Reminder Notification:** Reminder are sent to relevant parties through various channels, depending on the severity of the alert: - Healthcare Professionals:
Critical alerts are immediately sent to the healthcare provider or hospital, ensuring rapid intervention. –
Patients and Caregivers: High priority and low priority alerts can be sent to the patient's or caregiver's mobile app or web portal. This allows them to be informed about their health status in real-time and take appropriate actions.
6. **Actionable Insights:** Alongside alert notifications, the system can provide actionable insights, recommendations, and guidance to healthcare professionals, patients, and caregivers on how to respond to the alerts and manage the patient's health.
7. **Data Visualization:** Dashboards and visual representations of health data can help users track the patient's progress and make informed decisions.
8. **Historical Data:** The system stores historical health data, allowing for trend analysis, long-term monitoring, and retrospective assessments of the patient's health. In summary, a Smart Patient Health Monitoring System using IoT with an alert system plays a crucial role in proactive healthcare by continuously monitoring patients' health and notifying relevant stakeholders in real-time when critical or abnormal health conditions are detected. This timely information can significantly improve patient outcomes and enhance the quality of care.

**Components Required**
The following are the components required for making ESP8266 based Patient Health Monitoring System. All the components are easily available.

| S.N | COMPONENTS NAME | DESCRIPTION | QUANTITY |
|---|---|---|---|
| 1 | NodeMCU | ESP8266 12E Development Board | 1 |
| 2 | DS18B20 Sensor | DS18B20 One-Wire Waterproof Temperature Sensor | 1 |
| 3 | DHT22 Sensor | DHT22 Digital Humidity Temperature Sensor | 1 |
| 4 | Pulse Oximeter Sensor | MAX30100 I2C Pulse Oximeter Sensor | 1 |
| 5 | Jumper Wires | Male to Male Jumper Wires | 8 |
| 6 | Breadboard | Solderless Breadboard MIni | 1 |
| | | | |

**MAX30100 Pulse Oximeter Sensor**
MAX30100 is an integrated Pulse Oximetry and Heart Rate monitor sensor solution. It requires a 1.8V to 3.3V power supply to operate. Also, we can power it down through program code by decreasing its standby current and providing a power supply all the time. This oximeter sensor has two LEDs, photodetector optimized optics, and low-noise-analog signal processing to detect heart-rate signals.
The MAX30100 is a pulse oximeter and heart-rate sensor module that is commonly used in various applications, including wearable health and fitness devices. It is designed to measure two primary parameters: oxygen saturation (SpO2) and heart rate (HR).

Here is some key information about the MAX30100 Pulse Oximeter Sensor:
1. Principle of Operation: - The MAX30100 uses a combination of red and infrared (IR) light-emitting diodes (LEDs) and a photodetector to measure oxygen saturation and heart rate.
- The red LED is primarily used to measure heart rate, while the IR LED

is used to measure both heart rate and oxygen saturation.
2. **Oxygen Saturation (SpO2) Measurement: -** SpO2 is a measure of the oxygen saturation level in the blood, specifically, the percentage of oxygenated hemoglobin in the blood.
- The MAX30100 calculates SpO2 by measuring the absorption of red and IR light as it passes through a pulsating blood vessel.
3**. Heart Rate (HR) Measurement: -** The MAX30100 measures heart rate by detecting the variations in the amount of light absorbed by blood vessels due to the pulsatile flow of blood.
- It uses these variations to calculate the beats per minute (BPM), providing an accurate measure of heart rate.
4. Sensor Integration: - The MAX30100 integrates the LEDs, photodetector, and analog front-end processing into a single module, making it convenient for integration into various applications.
5. Communication Protocol: - The sensor typically communicates with a microcontroller or microprocessor via an I2C (Inter-Integrated Circuit) interface, making it easy to interface with a wide range of microcontroller platforms.
6. Sampling Rate and Resolution: - The MAX30100 supports a variety of sample rates and resolutions to accommodate different application requirements.
7. Applications: - Health and fitness wearables: The sensor is commonly used in smartwatches, fitness bands, and other wearable devices to monitor the wearer's SpO2 and heart rate.
- Medical devices: It can also be integrated into medical devices for continuous monitoring of patients' vital signs.
- Sports and fitness equipment: The sensor can be used in exercise equipment to provide real-time heart rate monitoring.
- Research and development: Researchers and developers often use the MAX30100 for physiological monitoring and experimentation.
8. Accuracy and Calibration: - The accuracy of SpO2 measurements can be affected by various factors, including skin pigmentation, ambient light, and sensor placement. Calibration may be required in some applications to ensure accuracy.
9. Power Consumption: - The MAX30100 is designed to be power-efficient, making it suitable for battery-powered devices.

The MAX30100 Pulse Oximeter Sensor is a valuable tool for monitoring vital signs and can be used in a wide range of applications, from consumer wearables to medical devices and research projects. Its integration of SpO2 and heart rate monitoring in a compact module makes it a popular choice for developers and designers in the healthcare and fitness industries.


**Working Principle -MAX30100 Pulse Oximeter**
The MAX30100 Pulse Oximeter has two built-in LEDs, in which one emits red light and the other emits infrared light. To measure pulse rate, only infrared light is required. But both red light and infrared light are used to measure oxygen saturation (Sp02) levels in the blood. When the heart pumps the blood, there is more oxygenated blood and when the heart rests, the amount of oxygenated blood also gets decreased. Hence, by knowing the time between the rise and fall of oxygenated blood, we
determine the pulse rate.

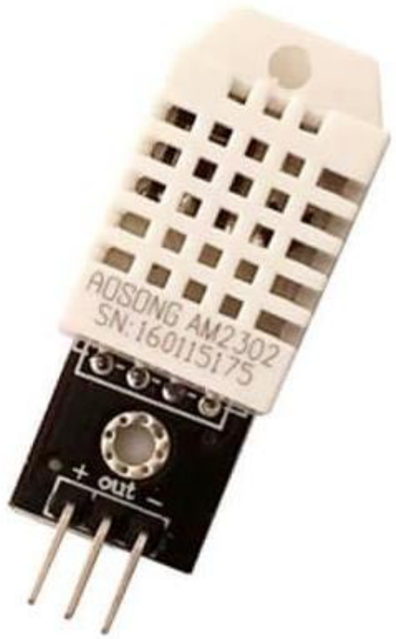
**DS18B20 Temperature Sensor**
This is a pre-wired and waterproof version of the DS18B20 sensor. This sensor is useful for measuring temperature from -55°C to 125°C (-67°F to +257°F) even in wet conditions. It has a long wire, so it's useful when a patient is a little far. Actually, the cable of this sensor is jacketed in PVC.

DS18B20 is a digital sensor, so there is no signal degradation in long distances. It is fairly precise, i.e. ±0.5°C over much of the range. This sensor works great with any microcontroller using a single digital pin. The downside is it uses the Dallas 1-Wire protocol, which is complex and
requires a bunch of code to communicate.

Volume: 08 Issue: 03 | March - 2024 SJIF Rating: 8.176 ISSN: 2582-3930

**DHT22 Temperature & Humidity Sensor**

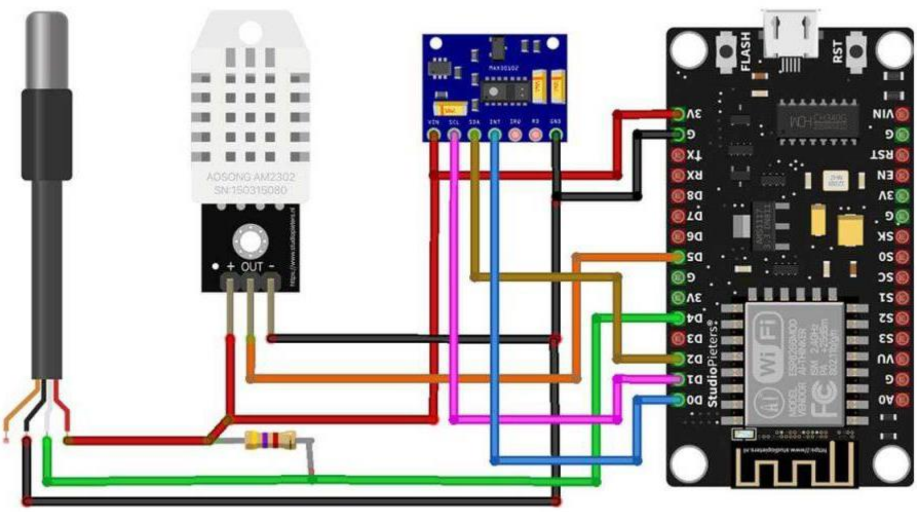
The DHT22 is a simple, ultra-low-cost digital temperature & humidity sensor. DHT22 uses a capacitive humidity sensor and a thermistor to measure the surrounding temperature and humidity. It sends data in digital signal form so no analog input pin is required.




This DHT22 Sensor provides more précised data than DHT11 Sensor. Alternatively, we can use the DHT11 Sensor. Learn more about DHT11 Humidity/Temperature Sensor.

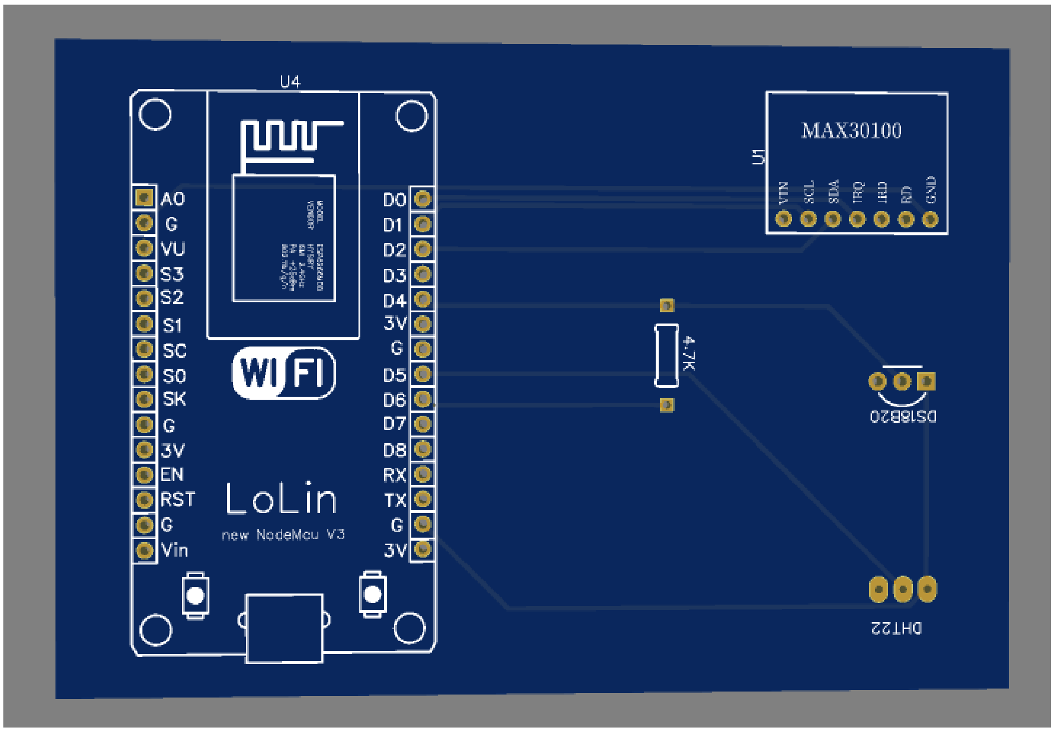
**Circuit Diagram: ESP8266 based Patient Health Monitoring System**

Now let us design IoT Based Patient Health Monitoring with ESP8266 Web Server. So, the circuit diagram for interfacing MAX30100, DHT22 & DS18B20 with ESP8266 is given below.



**PCB Design & Assembly**

This circuit requires a custom PCB because the circuit assembled on breadboard looks messy and isn't portable too. So, I have designed a custom PCB for this project.




**Source Code/Program Explanation**

The Program code for the ESP8266 Patient health monitoring system starts by including the following libraries: ESP8266WebServer.h library is used for connecting ESP8266 board to Wi-Fi network and setting up a webserver. Wire.h library is for communicating any I2C device not just the MAX30100 Pulse Oximeter sensor. MAX30100_PulseOximeter.h for reading BPM and Sp02 from the oximeter sensor. One Wire.h and Dallas Temperature.h library for reading data from the DS18B20 temperature sensor. Finally, DHT.h for grabbing Humidity and Temperature from DHT11/DHT22 sensor.

ESP8266 WebServer LibraryMAX30100 Library One Wire Library Dallas Temperature LibraryDHT Library

[redacted]

Here we defined the DHT sensor type, its signal pin interfaced with NodeMCU. Similarly, Dallas Temperature DS18B20 sensor pin and reporting period of 1000ms for MAX30100 sensor is also defined.

[redacted]

Five different variables (temperature, humidity, BPM, SpO2, and body temperature) are also defined.

[redacted]

Change your WiFi Network Credentials like WiFi SSID and Password here.

[redacted]

Initialize DHT sensor, Pulse Oximeter sensor, and DS18B20 Dallas Temperature sensor.

[redacted]

Start the webserver on ESP8266 NodeMCU module on port 80.

[redacted]

Begin serial debugging at a baud rate of 115200. Define NodeMCU D0 pin (GPIO 16) as output. Test the DHT 22 sensor and connect your microcontroller to the Wi-Fi network. After a successful connection, it will provide your IP address. Finally, Start the HTTP server and Initialize the MAX30100 sensor for testing and print results on the serial monitor.

[redacted]

```
} else {
Serial.println("SUCCESS");
}
```

Request sensor readings from all the sensors and print those five parameters (temperature, humidity, BPM, Sp02, and body temperature) on the serial monitor.

```
server.handleClient();
pox.update();
sensors.requestTemperatures();
if (millis() - tsLastReport > REPORTING_PERIOD_MS) {
float t = dht.readTemperature();
String Temperature_Value = String(t);
float h = dht.readHumidity();
String Humidity_Value = String(h);
bodytemperature = sensors.getTempCByIndex(0);
temperature = t;
humidity = h;
BPM = pox.getHeartRate();
SpO2 = pox.getSpO2();
Serial.print("Room Temperature: ");
Serial.print(t);
Serial.println("°C");
Serial.print("Room Humidity: ");
Serial.print(h);
Serial.println("%");
Serial.print("BPM: ");
Serial.println(BPM);
Serial.print("SpO2: ");
Serial.print(SpO2);
Serial.println("%");
Serial.print("Body Temperature: ");
Serial.print(bodytemperature);
Serial.println("°C");
Serial.println("*********************************");
Serial.println();
tsLastReport = millis();
}
```

If there is a successful connection, we can send these parameters to the ESP8266 local webserver.

```
void handle_OnConnect() {
server.send(200, "text/html", SendHTML(temperature, humidity, BPM,
SpO2, bodytemperature));
}
void handle_NotFound() {
server.send(404, "text/plain", "Not found");
}
```

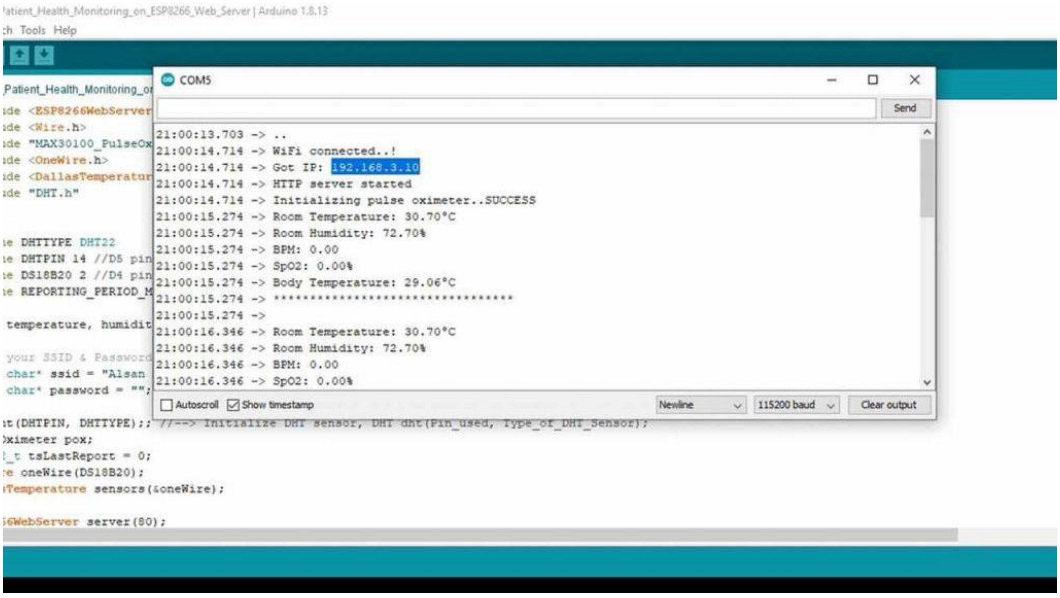
**Design Web Page with HTML and CSS**
We start the html code using an html string variable to display patient health monitoring parameters on the webpage dynamically.

```
String SendHTML(float temperature, float humidity, float BPM, float
```

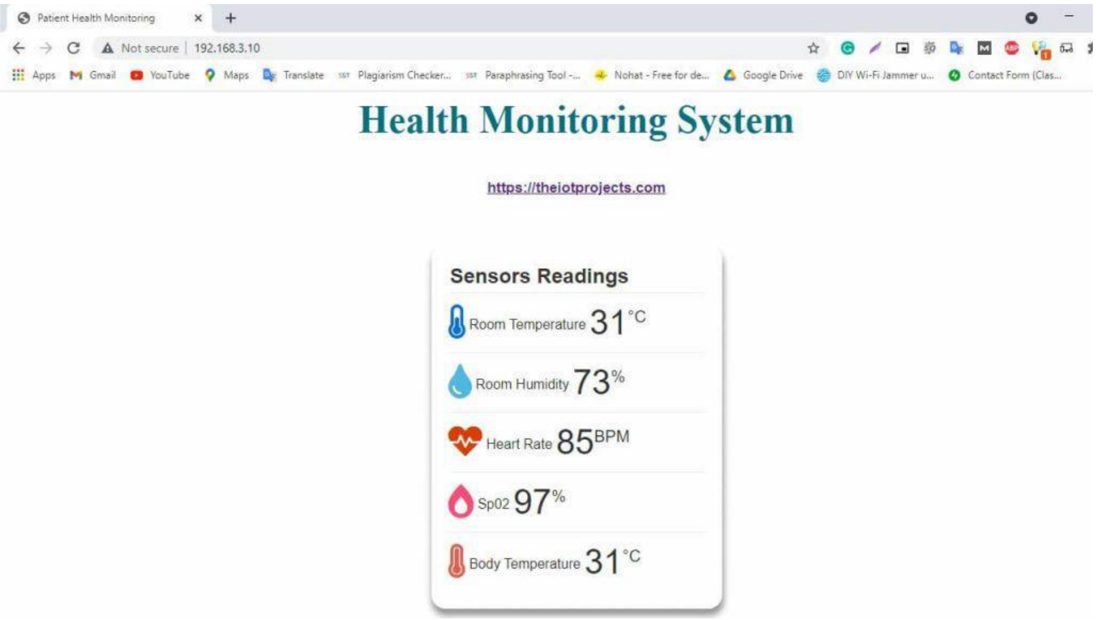
[redacted]

**Project Demonstration**

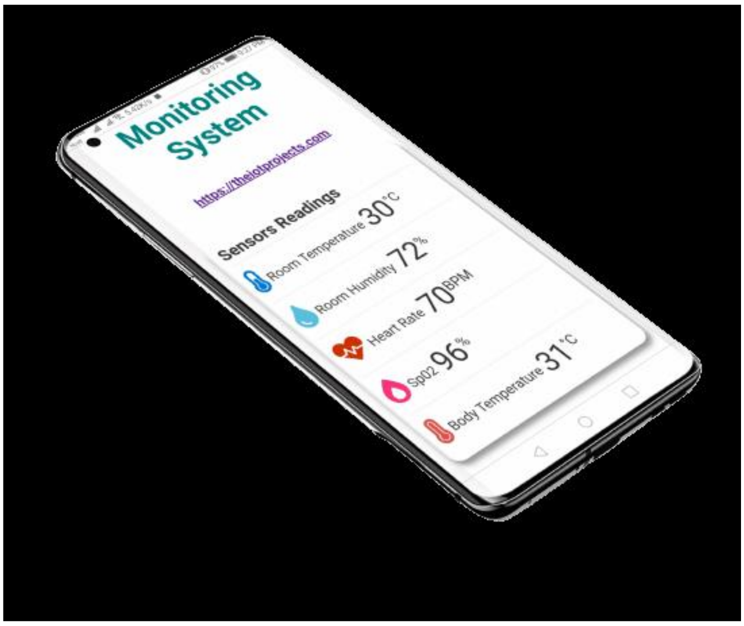
Once the code is uploaded to your NodeMCU ESP8266 board, you can open the serial monitor to see the program into action. The NodeMCU ESP8266 will connect to your Wi-Fi Network. Once connected, it will display the ESP8266 IP Address.




Now, copy the ESP8266 IP Address and paste it on your Web Browser. It will display the room temperature, room humidity, Heart Rate, Blood Oxygen Level, and body temperature, etc., as shown in the images below.




Similarly, you can monitor your patient's health from any device that features browsing capability. The below image is the view of the Patient Health Status on Android Smartphone. You simply need to copy the IP Address and paste it on the browser of any device.




## Conclusion

In conclusion, the development and implementation of a Smart Patient Health Monitoring System using IoT represent a significant leap forward in the realm of healthcare technology. This innovative system seamlessly integrates data collection, transmission, analysis, and alert generation to provide real-time insights into a patient's health status. By leveraging advanced algorithms and machine learning models, it can detect critical health events and deviations from normal parameters, enabling healthcare professionals, patients, and caregivers to respond promptly. The alert system, a pivotal component of this solution, categorizes alerts based on severity, ensuring that critical situations receive immediate attention while allowing for proactive management of less urgent concerns. Healthcare professionals can make informed decisions and intervene rapidly in life-threatening scenarios, potentially saving lives.

Patients and caregivers are empowered with timely information and actionable insights, enabling them to actively participate in their healthcare journey and make informed decisions about their well-being.

The ability to visualize health data through intuitive dashboards and access historical data for trend analysis enhances the overall effectiveness of the system. It provides a comprehensive view of a patient's health trajectory, aiding in long-term monitoring and retrospective assessments.

As we move forward in the era of digital healthcare, the Smart Patient Health Monitoring System using IoT stands as a prime example of how technology can revolutionize patient care. Its potential to improve patient outcomes, enhance early disease detection, and promote overall well-being cannot be overstated. However, it is essential to ensure the security and privacy of patient data, comply with regulatory requirements, and continually refine the system to incorporate the latest advancements in IoT and healthcare technologies. In the coming years, we can anticipate further refinement and widespread adoption of such systems, ultimately leading to a more patient-centric and proactive approach to healthcare. The journey towards a healthier future has begun, and the Smart Patient Health Monitoring System using IoT is at the forefront of this transformative wave.