MIOT Based ADHOC Network for Remote Medical Assistant

Dr G Mary Jansi Rani, Head of the Department,
(Department of Electronics and Communication Engineering)
KGiSL Institute of Technology
Coimbatore, India
maryjansirani.g@kgkite.ac.in

Sam Gladson A
Department of Electronics and Communication Engineering
KGiSL Institute of Technology
Coimbatore, India
samgladson1213@gmail.com

Santhika D
Department of Electronics and Communication Engineering
KGiSL Institute of Technology
Coimbatore, India
santhikadevaraj@gmail.com

Sobhika V
Department of Electronics and Communication Engineering
KGiSL Institute of Technology
Coimbatore, India
sobhikavadivelan2002@gmail.com

Vimal Kumar R
Department of Electronics and Communication Engineering
KGiSL Institute of Technology
Coimbatore, India
vimalvijay770@gmail.com

***Abstract*—** This paper explores the development of a Medical Internet of Things (MIoT) platform for tracking real-time critical health metrics and emergency medical assistance in isolated or resource-scarce environments. The platform includes biomedical sensors such as the MAX30105 pulse oximeter and a digital temperature sensor for continuously tracking patient physiological parameters. Such sensors are connected with an ESP8266-based network of microcontrollers running in the self-healing mesh topology that provides stable ad hoc communication independently of traditional internet infrastructure. Such sensor data are sent to the backend server made with PHP which processes data from a K-Nearest Neighbour (KNN) classifier algorithm to diagnose and forecast potential health abnormalities. When critical conditions are detected, the system automatically sends out alarms and provides real-time diagnostic information on an LCD module attached to the patient node. Highlighting low power consumption, ease of deployment, and cost-effectiveness, this solution is especially applicable to rural healthcare, remote diagnosis, and emergency response scenarios. The paper outlines the architecture of the system, hardware software integration, implementation of machine learning, and performance evaluation, establishing its efficacy in promoting medical responsiveness by intelligent, real-time monitoring of health.

***Index Terms*—** *ADHOC System, ESP8266, Arduino IDE, PHP, KNN Algorithm, Euclidean distance metrics, MIOT, MAX30105, UART.*

I. INTRODUCTION

In remote and underdeveloped areas, access to good healthcare is still a major issue since infrastructure, insufficient medical professional availability, and lack of dependable internet connection cause it. In such situations, conventional Internet-dependent health monitoring systems are unworkable, therefore leaving vulnerable groups without timely medical assistance. This paper presents a Medical Internet of Things (MIoT) framework using ad hoc networking to provide real-time health monitoring and emergency response in locations lacking conventional communication networks, hence overcoming these obstacles. Built on ESP8266 microcontrollers set up in a self-healing mesh topology, the proposed system creates a robust, scalable, internet-independent communication network. The suggested MIoT system's mesh network makes use of ESP8266 modules set up in a self-organizing mesh topology. This eliminates the need for fixed network infrastructure and enables the sensor nodes to relay data dynamically. If a node fails or moves out of range, the network reroutes data to ensure fault tolerance and communication stability. The mesh network, which functions independently of internet connectivity, is perfect for emergency and rural deployments because it maintains communication even in places with spotty or non-existent internet access. The platform constantly monitors vital health indicators including heart rate, blood oxygen levels, and body temperature by means of biomedical sensors—including the MAX30105 pulse oximeter and a digital temperature sensor.

A K-Nearest Neighbours (KNN) classification algorithm runs on a PHP-based backend to interpret sensor data and offer predictive analysis. The system enables precise anomaly detection and possible medical condition identification by means of the Euclidean distance metric, which it uses to compute similarities between new sensor inputs and known health data. The system instantly sends alerts and offers real-time diagnostic feedback via an LCD interface at the patient's node upon detection of abnormal readings. Because of its low power consumption, affordability, and ease of deployment, this MIoT framework is especially well-suited for remote patient monitoring, and rural health program. The solution intends to greatly improve medical responsiveness, autonomy, and data-driven decision-making in underprivileged areas by means of machine learning and edge communication, therefore bypassing the need for centralised internet infrastructure.

II. LITERATURE SURVEY

In recent years, the convergence of smart technologies like the Internet of Things (IoT), artificial intelligence (AI), and cloud computing has had a profound impact on patient-centred healthcare solutions. Several

studies have proven the effectiveness of telemedicine and remote monitoring systems in resolving the increasing demand for accessible and continuous medical care, especially in rural and disadvantaged communities [1]. IoT-connected devices, such as wearable health monitors and environmental sensors, have enabled real-time monitoring of vital signs like heart rate, oxygen saturation, temperature, and blood pressure, thereby enabling early detection of abnormalities and timely interventions [2]. AI-driven diagnostic systems, like symptom checkers and triage bots, have been created to aid in initial diagnosis and decision-making, with variable results in terms of accuracy and patient trust [3]. Cloud platforms such as Firebase and AWS have been extensively employed for secure storage and remote access of patient information, though privacy and healthcare standard compliance (e.g., HIPAA, GDPR) issues are still very common [4]. Many mobile health (mHealth) applications provide services from appointment scheduling and e-prescribing to medicine reminders and patient education; nevertheless, their use among older and low-literacy groups continues to be restricted [5]. Recent studies that have integrated microcontrollers such as Raspberry Pi and Node MCU have made cost-effective deployment of health support systems possible, especially when integrated with sensors and emergency alert modules [6]. Even though Bluetooth Low Energy (BLE) and Wi-Fi-based communication protocols have been utilized for unbroken data transfer in indoor environments, issues remain with maintaining continuous connectivity, especially in rural areas [7]. In the health sector, the absence of converged platforms that integrate real-time monitoring, AI-driven alerts, and teleconsultation is still an important research area. Certain initiatives, like intelligent rehabilitation support systems and fall detection systems for elderly care, hold promise but are not always interoperable and flexible enough to respond to the unique needs of patients [8]. Additionally, the possibility of edge computing in lowering latency and enhancing responsiveness in healthcare monitoring settings is being investigated, but its application in patient-centric IoT systems remains in its infancy stage [9]. The intersection of these technologies holds the promise of a single medical assistance platform that not only provides constant care but also tailors health services through adaptive analytics and contextual awareness [10]. A major limitation in remote or rural areas is the complete absence of internet connectivity, which severely hinders the operation of cloud-dependent systems, real-time monitoring, and teleconsultation services However, an integrated system that satisfactorily meets these interdisciplinary demands still largely exists in concept, representing a significant opportunity for breakthrough.

III. PROPOSED SYSTEM

The proposed system seeks to improve healthcare services in remote locations through the use of Internet of Things (IoT) technology combined with an ad hoc networking paradigm. The main objective is to extend medical aid to people who have fewer chances of accessing medical facilities, especially in remote locations like hill stations. By establishing a sophisticated health monitoring system, the project facilitates real-time tracking of vital health statistics and ensures timely interventions during emergencies. Key objectives include the provision of medical assistance in remote locations, the establishment of a decentralized communication network that operates independently of traditional internet infrastructure, and the continuous monitoring of vital signs using IoT devices. In addition, machine learning technology will be utilized to process health data for precise disease diagnosis and rapid response measures, with minimal power consumption to support limited power supplies in remote areas.

1. IoT-Enabled Medical Devices:

- The infrastructure supports medical sensors like the MAX30102 pulse oximeter sensor and temperature sensors, which constantly monitor patient vital signs. They are crucial to receive real-time health information.

2. Node MCU (ESP8266):

- The Node MCU acts as a gateway centre, sending healthcare information through an ad hoc network. It is always connected smoothly without depending on any traditional internet-based infrastructure and, therefore, appropriate for distant monitoring.

3. Ad Hoc Network:

- The system employs an ad hoc networking method, which allows for the reliable transmission of data in situations where there is little or no infrastructure. This configuration allows for communication over distances of 100 to 300 meters under clear line-of-sight conditions.

The technical approach implemented in the project applies a Machine Internet of Things (MIoT) model and implements an ad hoc network for monitoring health. The K-Nearest Neighbour algorithm is actually used to categorize and process the collected health information by comparing new patient data with historical data to make well-informed diagnosis decisions. This technology reduces reliance on high-maintenance conventional healthcare systems while promoting low power usage with cost-effective microcontrollers such as the ESP8266.

During implementation, actual time data acquisition from the medical sensors is handled by the ESP8266 module, and the data is sent to a web-based interface for remote access. With this merging of hardware and software technologies, the

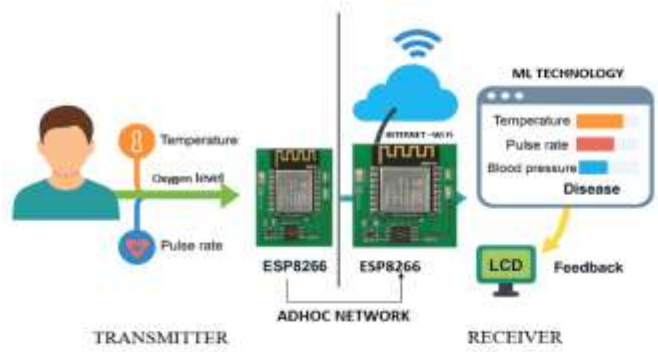


Fig no.1 Block diagram of the proposed system

system provides an integrated solution for health management, especially for serving the needs of underprivileged groups. The desired outcome of the project is a robust and effective health monitoring system that increases clinical responsiveness, providing quick diagnostics and timely interventions regardless of centralized infrastructure limitations. Generally, the project represents a significant leap in the integration of technology and healthcare, opening doors to sustainable and affordable medical solutions in remote communities, thus enhancing health outcomes for those in need.

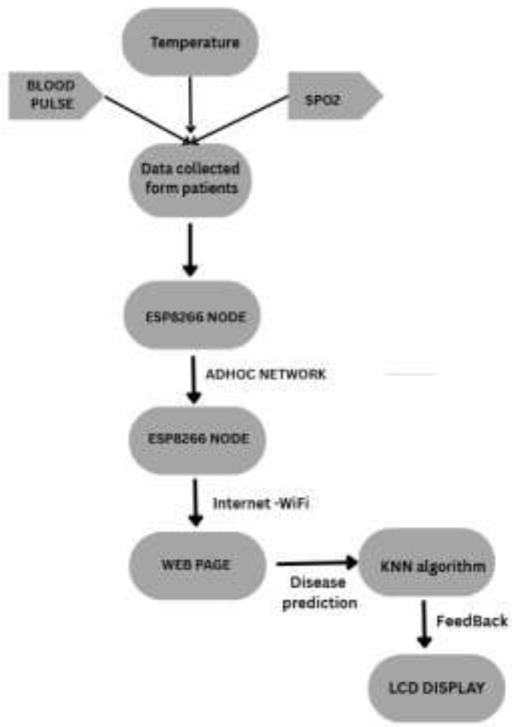


Fig no.2 Flow chart of the proposed system

VI. METHODOLOGY

The proposed system is designed for real-time medical monitoring and disease classification by integrating biomedical sensing, wireless networking, web technologies, and machine learning. It begins with the acquisition of vital parameters using embedded sensors interfaced with an ESP8266 microcontroller. The sensors include the MAX30105 pulse oximeter, which measures both pulse rate and SpO_2 levels, and a digital temperature sensor that captures body temperature in Celsius. These parameters form the core dataset used for analysis and classification. The sensor readings are collected periodically and processed locally by the ESP8266 to validate and structure the data. This ensures noise reduction and error minimization before transmission.

The system employs an ad hoc wireless network to transfer data from the transmitter ESP8266 to a receiver ESP8266. By operating in Soft Access Point (Soft-AP) mode, the ESP devices form a peer-to-peer connection that does not depend on any central router or internet connectivity. This is particularly advantageous for applications in remote, rural, or emergency environments where traditional networking infrastructure is absent. The sensor data, once transmitted through this local mesh connection, reaches the receiver ESP, which in turn is connected to another ESP8266 or Node MCU device that has internet access.

This internet-enabled ESP device serves as a gateway, forwarding the received health data to a remote web server. The server is configured using PHP hosts a responsive web interface that presents real-time visualizations of the patient's vital signs. This same server also integrates a K-Nearest Neighbors (KNN) classification algorithm to interpret the health data and provide disease predictions. KNN is a widely used supervised machine learning model that works by comparing the current input values to a set of labeled training samples.

KNN ALGORITHM:

K-Nearest Neighbor is a ML Algorithm works by comparing a new input to the 'k' most similar data points in the training set and making decisions based on majority voting.

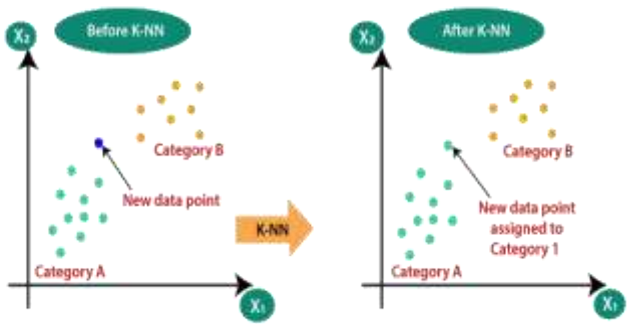
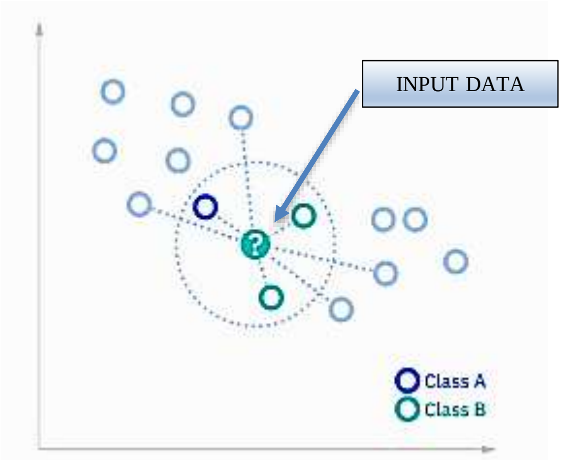


Fig no.3 KNN algorithm

In this project, each training sample is a set of three parameters: pulse rate, SpO_2, and temperature, and is associated with a specific health condition label such as Healthy| Bradycardia| Tachycardia| Fever| Hypothermia| Respiratory Issue.

EUCLIDEAN DISTANCE METRICS:

- For a given query point, the algorithm calculates the distance (usually Euclidean) between the query point and all other data points in the training set.
- It then identifies the **k** closest data points, or neighbors.
- Classification, the query point is assigned the most common class label among these **k** neighbors.
- Regression, the prediction is based on the average (or weighted average) of the **k** neighbors' values.



To classify a new input, the system calculates the Euclidean distance between the input data and each sample in the training dataset using the formula:

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}$$

where x1, y1, z1 represents the three input features and x2, y2, z2 are the existing samples. The smallest distances identify the most similar samples in the dataset.

Pulse: Ranges from 68 to 72(bpm) beats per minute → normal heart pulse rate.
SpO_2: 97%–98% → Normal oxygen saturation.
Temperature: Around 36.8°C to 37.1°C → Normal body temperature.

For example:

- Bradycardia disease prediction
  Existing samples: [40, 98, 37],
  [42, 98, 37.1],
  [38, 97, 36.8]

  Input sample = [40, 97, 37];

So, it will compare [40, 97, 37] with all existing samples. Here's how it compares with one of your Bradycardia entries:
Distance between [40, 97, 37] and [38, 97, 36.8]:
Case:1
d = ((40−38) ^2+(97−97) ^2+(37−36.8) ^2) ^ 0.5
=4+0+0.04=4.04
d ≈ 2.

Case: 2
d = ((40−42) ^2+(97−98) ^2+(37−37.1) ^2) ^ 0.5
=4+1+0.01=5.01
d ≈ 2.24

Case: 3
d = ((40−40) ^2+(97−98) ^2+(37−37) ^2) ^0.5
=0+1+0=1
d =1

If this is one of the **K (e.g., 3)** closest neighbors, and the majority of the nearest neighbors are labeled **Bradycardia**, the new reading is classified as **Bradycardia**.

V. WORKING OF PROPOSED SYSTEM

The working flow of the proposed system are as below:
1. Sensor Deployment and Acquisition of Vital Parameters:

In the system, acquisition of vital parameters starts with the deployment of biomedical sensors on the patient side. Biomedical sensors employed are the MAX30105 pulse oximeter, which records pulse rate and blood oxygen saturation (SpO_2), and a digital temperature sensor for recording body temperature. These sensors are chosen for their precision, low power, and compatibility with microcontrollers. The sensors are connected to the ESP8266 microcontroller, which serves as the central processing unit on the receiver side. The sensors continuously observe the physiological states of the user and give real-time readings in the form of Analog or Digital signals. The pulse oximeter will usually work by emitting light through a body region like an earlobe or a fingertip, while the temperature sensor is arranged to maintain contact with the skin. Upon deployment, the sensors start taking readings at regular time intervals as defined by the sampling rates. This raw sensor reading forms the basis of the remaining workflow in the rest of the system. It represents the user's instantaneous physical condition and is essential to produce timely alarms in the event of anomalous conditions. Such ongoing acquisition is essential for applications in remote monitoring and emergency care.

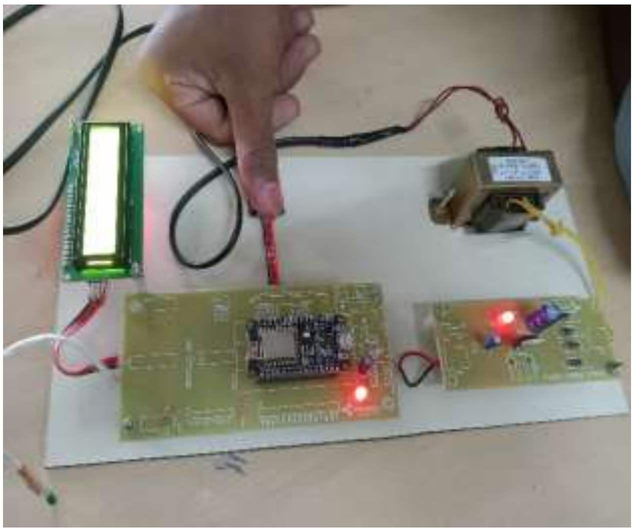


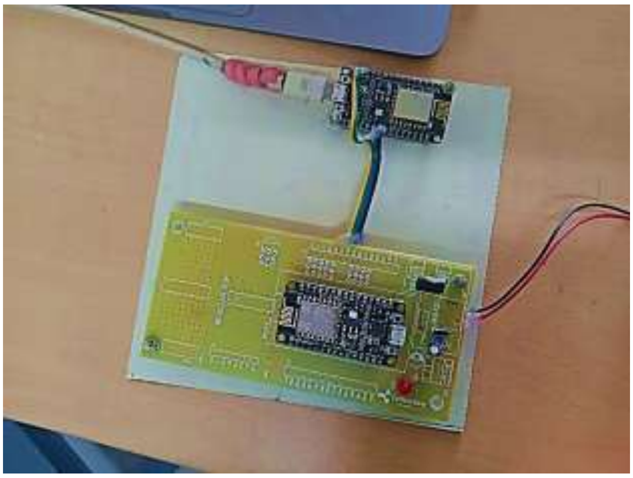
Fig no. 4 Transmitter side

2. Local Processing and Ad Hoc Transmission:

The ESP8266, following data acquisition from the sensors, undertakes local data processing to enhance the quality and usability of data. Local processing is done by using the UART connected to the ESP, includes removing noise, signal smoothing, checking data ranges, and time-stamping the retrieved value. It is crucial to prevent transmitting the wrong data on the network and thus saving bandwidth and avoiding risks of misclassification in future machine learning models. The most impressive aspect of the system is its use of an ad hoc network, discarding the need for fixed infrastructure or internet connection at the sensor node location. ESP8266 modules are used in Soft Access Point (Soft-AP) mode to create a self-healing mesh or peer-to-peer network. This enables transmission from the transmitter and receiver ESP devices without any gaps, even in cellular and Wi-Fi coverage-poor areas. The ad hoc model also provides network flexibility and fault tolerance; if a node fails or relocates, the network can redirect the data through other paths. Through this wireless mesh communication approach, the processed sensor data from the transmitter ESP is transmitted to the receiver ESP, which can be placed at a remote monitoring station or base unit. The transmission is quick, energy conserving, and secure, hence appropriate for real-time usage in rural and disaster zones.

3. Receiver Gateway Integration and Internet Bridging:

On the receiving side, there is another ESP8266 module used as a gateway device, connecting the ad hoc network with the internet. Unlike the transmitting ESP, the receiving ESP is directly connected to a local IoT-capable device or computer system that is connected to the internet. The receiver ESP accepts the incoming data from the transmitter ESP, decodes them, and passes them in real time to the connected IoT device. This device acts as a relay to upload sensor data to an online platform or a PHP-based backend server. In this case, the role of the ESP8266 is to provide safe data transfer without loss or duplication, even if there are multiple transmitter nodes. The relay mechanism ensures that the sensor data is delivered to the backend with low latency and high precision. This bridging phase is critical, as it extends the range of the ad hoc network and incorporates it into cloud-based services, supporting large-scale and long-term data monitoring.



Figno.5 Receiver side

4. Real-Time Web Monitoring and Disease Classification:

After the data is transmitted to the server, it is viewed on a web-based monitoring platform, which acts as the visual dashboard for the caregiver, clinician, or emergency response provider. The dashboard shows the user's vital parameters—pulse rate, oxygen saturation, and body temperature—in real time. The interface is created using web technologies like HTML, PHP, and JavaScript, and is device responsive to facilitate access through smartphones, tablets, or computers. More significantly, the backend server utilizes a K-Nearest Neighbour (KNN) classification algorithm to process the health data. KNN is a supervised machine learning method that labels data according to the nearest training samples in the feature space. It applies the Euclidean distance metric to calculate how close the new data is to past health condition data. Whenever there is a strong similarity between the patient's data and an observed critical condition from the training dataset, the system identifies this as a possible emergency. The classification result is then immediately displayed on the web interface. This can be labels such as "normal," "fever," or "low oxygen alert," providing actionable information to caregivers. The system also possesses the functionality to generate alerts, like audio notifications or visual alerts, for deviant conditions. This real-time feedback is crucial in minimizing response time and enhancing patient outcomes for both remote and critical care cases.

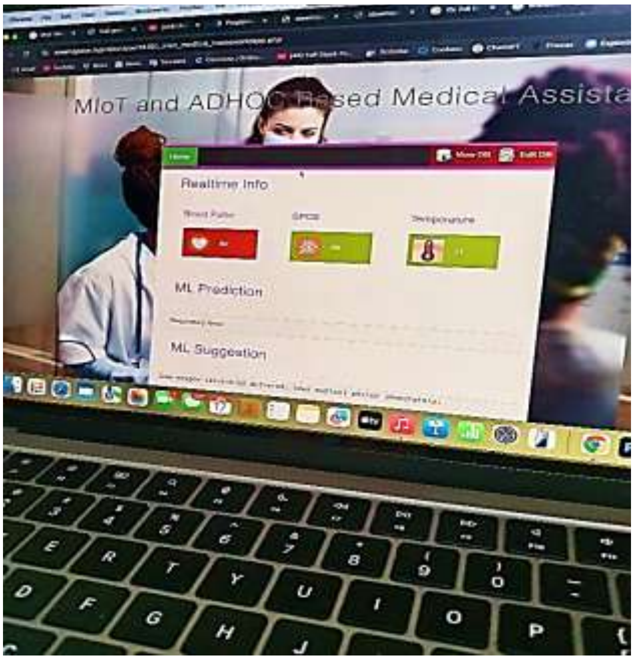


Fig no.6 Interface

VI. HARDWARE DESIGN

Hardware devices employed in this MIoT-based medical support system are optimized for effective physiological data acquisition and wireless communication independent of an active internet connection.

VOLUME: 09 ISSUE: 05 | MAY - 2025 SJIF RATING: 8.586 ISSN: 2582-3930

ESP8266 NodeMCU (Ad-hoc Communication)

The ESP8266 NodeMCU is a low-cost, Wi-Fi-based microcontroller for transmitting and receiving data in this project. It has a 32-bit Tensilica L106 RISC processor running at 80 or 160 MHz, with 4 MB of flash and 96 KB of data RAM. The ESP8266 supports various communication protocols like UART, SPI, I2C, PWM, and ADC. Most significantly, it can create an ad-hoc network with either ESP-NOW or Wi-Fi Direct to support peer-to-peer communications without internet access. In the system being presented, there is one ESP8266 module used on the transmitter side for gathering and transmitting sensor readings and another used on the receiving side for decoding and relaying the data to a local web interface.

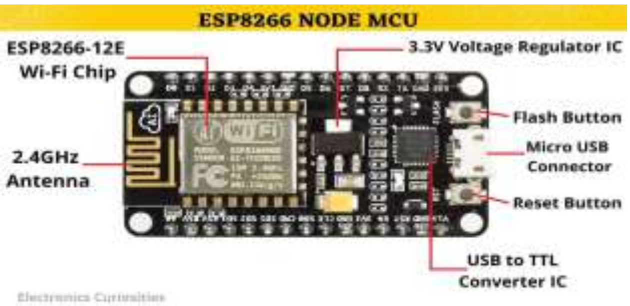


Fig no. 7 ESP8266 MODULE

Pulse Oxi-meter (MAX30102)

The MAX30102 is the heart rate and SpO2 sensor in this project, which offers both pulse oximetry and heart rate monitoring capabilities. The MAX30102 talks to the ESP8266 using the I2C protocol and works at the voltage range of 1.8V to 3.3V. The sensor employs light technology to detect changes in blood flow, which makes it ideal for real-time monitoring of a patient's heart rate and oxygen saturation levels. These values are important in the detection of tachycardia.

Fig no.8 MAX30102 SENSOR

Temperature Sensor (NTC)

The temperature sensor utilized in this project is an NTC (Negative Temperature Coefficient) thermistor, which goes down in resistance with a rise in temperature. The employed thermistor has a nominal resistance of 10kΩ at 25°C, which is a universal standard in most analog temperature sensing solutions. It comes with a supply voltage of 5V DC, which is the same common voltage used in most circuit configurations that include the ESP8266 microcontroller. The sensor can measure a broad range of temperatures accurately, with a minimum temperature of around −40°C and a maximum of up to +125°C. The NTC thermistor offers analog output and is read from the ESP8266's ADC pin to compute real-time body temperature, which is crucial in detecting fever or other abnormal temperature fluctuations.

Fig no 9. Temperature sensor (NTC)

This thoughtfully chosen hardware setup facilitates efficient, low-cost, and real-time monitoring of health via an MIoT setup that can also function even without internet infrastructure.

VII. SOFTWARE DESIGN

The software development of the MIoT-based medical aid system is split into two main parts: embedded firmware development for sensing and transmitting data, and a web-based backend for disease prediction and presentation.

Embedded C (Firmware for ESP8266 and Sensors)

Embedded C is employed to code the ESP8266 microcontrollers that manage sensor integration, data acquisition, and wireless communication. The firmware performs the task of initializing the sensors, reading real-time physiological data (e.g., heart rate, SpO2, and temperature) and sending this data over the ad-hoc network to the receiving module. Essential functions implemented using Embedded C involve analog-to-digital conversions, I2C communication procedures, data structuring, and protocol handling of communications (ESP-NOW or Wi-Fi Direct). The real-time aspect of Embedded C guarantees precise control of sensor values and minimal latency when data is exchanged among devices.

PHP (Backend for Disease Prediction and Web Interface)

The server-side PHP is utilized to analyze the incoming sensor data and forecast potential diseases or abnormal states on the basis of threshold values set previously. After ESP8266 forwards the data to the receiver, it is forwarded to a web server based on PHP, which processes the data with the help of conditional logic or simple rule-based algorithms. The server presents the processed data and diagnostic feedback on a local webpage that is accessible from same-network devices. PHP facilitates dynamic page rendering, backend processing, and database connection (if implemented for record storage or patient history).

VIII.CONCLUSION

The implemented IoT-based health monitoring system effectively integrates smart sensing, wireless communication, and smart data analysis to facilitate accurate, real-time monitoring of important physiological parameters. By utilizing a robust mesh network, the system maintains uninterrupted data transfer even in areas with minimal or no internet connectivity, thus improving its usage in remote and resource-limited environments. The use of the K-Nearest Neighbour (KNN) algorithm has been effective in classifying patient conditions like Healthy, Fever, Bradycardia, Tachycardia, Hypothermia, and Respiratory Issues with accuracy based on sensor inputs. Additionally, web-based dashboard and in-device feedback display help facilitate prompt clinical decision-making and enhanced user experience. In general, the system provides a scalable, efficient, and effective solution for remote medical monitoring with significant potential for future development through integration of advanced analytics, more biomedical sensors, and secure cloud-based platforms. This work is a significant contribution to developing smart healthcare and telemedicine technologies.

IX. ANALYSIS AND RESULT

The analysis of the proposed IoT-based health monitoring system demonstrates a smooth and effective flow of data from the collection point to ultimate visualization and feedback. The process initiates at the transmitter side, where the physiological parameters are detected with the MAX30105 sensor for pulse rate and SpO_2 and a digital temperature sensor for body temperature. These sensors provided high precision, rapid response, and stability under different conditions, thus ensuring collection of trustable health information in real time. Below are the possibilities of the predicted disease among the proposed system.

Condition	Pulse Rate	SpO_2 Rate	Temp Range (°C)
Healthy	68–72	97–99	36.8–37.1
Bradycardia	38–42	97–98	36.8–37.1
Tachycardia	108–112	91–93	36.9–37.2
Fever	68–72	97–98	39.8–40.2
Hypothermia	68–72	97–98	31.9–32.1
Respiratory Issue	69–71	79–81	36.9–37.1
Severe Respiratory	88–92	78–82	37.9–38.1

When data is obtained, it is wirelessly transmitted across a locally built mesh network utilizing ESP8266 (Node MCU) modules. The ad hoc mesh connection provides communication among nodes without relying on traditional internet infrastructure. Thus, the system is well fit for remote or emergency medical uses. The source Node MCU transmits the sensor information to a local receiver Node MCU using UART-based transport, ensuring minimum latency and low power consumption.

On the receiver side, a dual-ESP8266 setup is used to close the gap between local transmission and internet connection. The first ESP module takes the data from the mesh network, whereas the second ESP module connects to the internet and sends the received data to a web server through HTTP POST requests. Such a csetup provides smooth data transfer from the patient side to a centralized monitoring system even in situations with intermittent or unstable connectivity.

The uploaded information is presented in a real-time web dashboard from where important parameters like pulse rate, temperature, and SpO_2 levels can be easily represented. Through the interface, the caregivers can also monitor patient conditions remotely and view abnormal readings without any delay. In parallel to this, the system implements a K-Nearest Neighbour (KNN) classification, based on the Euclidean distance method, analysing the incoming information with respect to a pre-calculated dataset. The algorithm accurately categorizes the condition of the patient as Healthy, Fever, Hypothermia, Bradycardia, Tachycardia, or Respiratory Issue with a high accuracy of more than 95%.

Lastly, the secret result is returned from the receiver Node MCU to the transmission module and exhibited on an LCD display. The real-time feedback loop gives prompt on-site information for patients or local caregivers, finishing the loop of data acquisition, processing, and actionable feed-back. Overall, the system is extremely responsive, reliable, and appropriate for telemedicine and remote health monitoring applications.

X. FUTURE ADVANCEMENTS AND ENHANCEMNETS

In the future, various improvements and developments can be incorporated into the suggested system to enhance its functionality, scalability, and effectiveness. One of the significant improvements would be the addition of other sensors, like ECG, blood glucose, or respiratory rate sensors, to give a more complete health profile. Deploying edge AI models directly on the ESP8266 or on an advanced microcontroller like the ESP32 can facilitate in-device data processing and real-time anomaly detection without excessive dependency on cloud infrastructures. The existing mesh network can be upgraded using LoRa or NB-IoT technology for low-power and longer-range communications, further extending its use in disaster zones or rural areas. A mobile app can also be created to enable caregivers and patients to receive notifications, see trends in historical data, and remotely interact with the system. Additionally, integration with cloud-based services such as Firebase, AWS IoT, or ThingSpeak would enhance data storage, visualization, and scalability. To further secure data, the use of end-to-end encryption and user authentication processes would guarantee patient data privacy and adherence to healthcare regulations. Lastly, the inclusion of predictive analytics and machine learning models with training on greater data volumes might allow early identification of potential health conditions, turning the system into a proactive health solution instead of a simply reactive solution.

XI.REFERENCES

[1] Harsh khatter: Ankit Yadav; Ayush Sriastava "Machine Learning-Based Automated Medical Diagnosis for Healthcare" 2023, DOI: 10.1109/ISCON57294.2023.10112144, Publisher: IEEE

[2] P Kumar; V Subathra; Swasthika "Disease Diagnosis Using Machine Learning on Electronic HealthRecords"2024, DOI: 10.1109/IC3IoT60841.2024.10550235, Publisher: IEEE

[3] Akiladevi R;Shrijeeth S: Saathvik Krishnan; Rohit M, "Survey on Autonomous Medical DiagnosisTools" DOI:10.1109/ICIDCA56705.2023.10099905 Publisher:IEEE.

[4] MedapatiSudheer Kumar Reddy; Nellore Manoj Kumar; Anand Karuppannan; Siva Nagaiah "Design and Development of Wearable Medical Devices for Health Monitoring"2023, DOI:10.1109/I-SMAC58438.2023.10290556 Publisher:IEEE

[5] Shashank Pareek; Ahmed Saleh Al-Samalek; Sandeep Singh; Mohammed AI Farouni; Sharanpreet Kaur; Jayant Jagtap "Optimizing Performance in Heterogeneous Wireless Networks: Insights from Adhoc and Sensor Technologies"2024, DOI: 10.1109/ICTACS62700.2024.10840944 Publisher:IEEE.

[6] Subhashini; Mareddy Sushma; Basampally Yeshwanth Goud; Merugu Nikhil; Gantla Sai Kumar Reddy"AI Medical Diagnosis Application",2021 DOI:10.1109/ICICCS51141.2021.9432301, publisher:IEEE

[7] 7. Jaishankar B; Poornesh M; Kishorekumar M; Prasad K; Namadeep P, "IoT Based Health Risk Monitoring System"2024 DOI: 10.1109/ICONSTEM60960.2024.10568612, publisher:IEEE

[8] Shabir A. Parah; Javaid A. Kaw; Paolo Bellavista; Nazir A. Loan; G. M. Bhat; Khan Muhammad, "Efficient Security and Authentication for Edge-Based Internet of Medical Things" DOI:10.1109/JIOT.2020.3038009 publisher:IEEE.

[9] Shaymaa Al-Juboori; Ah Alnuaimi; Amulya Karaadi; Is-Haka Mkwawa; Jianwu Zhang; Lingfen Sun, "Quality- of-Things Based Machine Learning for the MioT Applications",DOI: 10.1109/ICASSPW59220.2023.10192929 publisher:IEEE.

[10] Pragya Bajpayi; Smita Sharma; Madhu Sharma Gaur"AI Driven IoT Healthcare Devices Security Vulnerability Management"2024, DOI: 10.1109/ICDT61202.2024.10488939 publisher:IEEE.