

## RESEARCH ARTICLE



# Edge Computing-Based Cardiac Monitoring and Detection System in the Internet of Medical Things

Saranya Murugesan<sup>1,\*</sup> and Archana Naganathan<sup>2</sup>

<sup>1</sup>Department of Instrumentation and Control Systems Engineering, PSG College of Technology Coimbatore, India

<sup>2</sup>Department of Electrical and Electronics Engineering, PSG College of Technology Coimbatore, India

**Abstract:** Currently, cardiac disease becomes one of the major health issues in the world. The death due to cardiac arrests has been increasing day by day. The survival rate for the sudden cardiac arrest is low. It is possible to save 60% human life by continuously monitoring the fitness of the person. In recent times, the Internet of Things (IoT) is emerging promising technologies. If the doctor cannot monitor the patient face to face, there IoT plays important role in remotely monitoring the patient's condition. With the rise in the use of smart wearable gadgets, the contribution of IoT will be more. In this paper, about five predictor models are analyzed for a particular dataset and chosen the best as Random forest algorithm. The proposed system integrates oxygen saturation level and heart rate sensor data from MAX30100 sensor for monitoring the patient's vital signs. The remaining parametric data of the patient (AGE, GENDER, BLOOD SUGAR LEVEL, BLOOD PRESSURE LEVEL, GENETIC ISSUE, and CHLOESTEROL LEVEL) are fed through the matrix keypad. The designed system monitors the online patient's vitals continuously. The patient's vitals will be stored in the edge server and also analyzed. The chosen predictor model checks the vital data for any abnormality and based on abnormality data detection in the heart rate, and the warning message is conveyed to the nearby health care center via ThingSpeak cloud. By tracking the patient's GPS location, the server cloud sends alert notification along with the GPS location to the nearby health center and to the registered mobile numbers.

**Keywords:** heart rate, random forest algorithm, cloud server

## 1. Introduction

In recent years, cardiac disease becomes one of the major health issues in the world. The death due to cardiac arrests has been increasing day by day. The survival rate for the sudden cardiac arrest is low. It is possible to save 60% human life by continuously monitoring the fitness of the person [1]. In recent times, the Internet of Things (IoT) is emerging as one of the promising technologies. The internet's accessibility and the advancement of technology have made it possible for many devices to interact and exchange information. IoT is another concept that allows customers to interact with a variety of smart devices and sensors for gathering data in real time from the environment [2]. However, a significant amount of knowledge needs to be gained from the perspective of e-Health as well as m-Health models for the purpose to identify and communicate vital information associated with a patient's health through the use of mobile phone sensors. If the doctor cannot monitor the patient face to face, there IoT exhibits an important role remotely in monitoring the patient's health [3]. With the rise in the use of smart wearable gadgets, the contribution of IoT will be more. Many patients die as a result of ignorance and inadequate medical care provided to them when they need it most. This is a result of

the doctor's inability to continuously observe the patient. Reducing the frequency of hospital visits by elderly individuals can lower healthcare expenses and ease the burden on the healthcare system as a whole. By using cardiac monitoring system, IoT helps to track, monitor, and alert the person [4, 5].

In this paper, our proposed system incorporates heart rate sensor and temperature sensor for monitoring the patient's vital signs. This system monitors the patient's vitals continuously. If any abnormality is detected, i.e., exceeds the threshold value in this monitoring process, the alert message is sent via e-mail to family members and hospital from cloud server [6, 7]. So, the data collected during the abnormality are sent to the centralized server cloud.

### 1.1. Proposed system

The purpose of cardiac monitoring system is to give real-time early warning message by monitoring the oxygen saturation rate and heart rate. The oxygen saturation rate and heart rate are monitored by using MAX30100 sensor. Any deviation from a usual state of health will be reported by e-mail to the hospital and the family for additional investigation and preventive measures. Patients who suffer from heart attacks are spared death as a result. The suggested methodology's schematic representation is displayed in Figure 1. The cardiovascular monitoring system's block diagram along with the emergency alarm are displayed in Figure 2.

\*Corresponding author: Saranya Murugesan, Department of Instrumentation and Control Systems, PSG College of Technology Coimbatore, India. Email: [msa.ice@psgtech.ac.in](mailto:msa.ice@psgtech.ac.in)



Figure 1. Schematic representation of proposed methodology

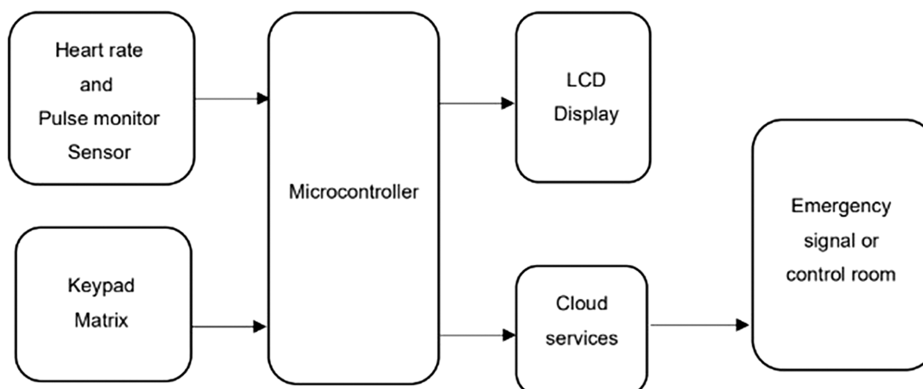


Figure 2. Block diagram of cardiac monitoring system and emergency alert

It utilizes the sensor to measure oxygen saturation and heart rate. Sensor and matrix keypad are interfaced to the Raspberry Pi that allows checking heart rate readings and transmitting them over to cloud [8]. When one of the fingers is placed on the sensor’s surface, a digital representation of the heartbeat is produced.

The LED flashes when the sensor activates. Beats per minute, or BPM, is the output generated. After setting high and low limits, the system starts monitoring. If these parameters go to abnormal, it will automatically send alert message to the emergency room.

## 2. Design Methodology

The design portion is mostly divided into two sections:

- Architectures for hardware
- Architecture for software

### 2.1. Architecture for hardware

It comprises MAX30100, Raspberry Pi 3 model B, Matrix keypad, and LCD Display.

#### 2.1.1. Raspberry Pi 3 model B

With a 64-bit quad-core processor operating at 1.4 GHz, dual-band 5 GHz, and 2.4 GHz wireless LAN, Bluetooth 4.2/BLE, improved Ethernet, and PoE functionality via a separate PoE HAT, the Raspberry Pi 3 Model B+ is the newest member of the Raspberry Pi 3 family. Because of the dual-band wireless LAN’s modular compliance certification, there is less expense and time to market since the board may be integrated into end products with substantially less wireless LAN compliance testing. The Raspberry Pi 3 Version B+ has the same mechanical footprint as the Raspberry Pi 2 B model and the third-generation Raspberry Pi Version B [9].

#### 2.1.2. Heart rate MAX30100 sensor

A piece of equipment that combines heart rate monitoring and pulse oximetry is called the MAX30100. With the use of two LEDs, a photodetector, better optics, and low-noise analogue signal processing, it can detect heart rate and pulse oximetry signals. This sensor is connected to the other microcomputer via the I2C interface [10, 11]. Wear this device on the fingertips or earlobe with the purposes of measuring blood oxygen levels and heart rate. Any non-thick part of the body may also wear it. Both infrared radiation and red light can pass through your finger’s tissues when you place it there for measurement, and a photodiode may pick up on their absorption [12]. The amount of captured red light divided by the amount of infrared light will vary.

#### 2.1.3. Matrix keypad

These Keypad modules are made of thin, flexible membrane material. The 4 × 4 keypad module’s 16 keys are arranged in a pattern of rows and columns. Each of these switches is connected to the other via a copper connection. Most of the time, rows and columns are not connected. A series of rows and a series of columns come together whenever we push a key. A 4 × 4 keypad will have eight terminals. There are four matrix rows and four matrix columns in each of them. The module has 16 buttons, which are used to drive these 8 pins. The 16 buttons grouped in a matrix arrangement are represented by the 16 alphanumeric digits on the module surface.

#### 2.1.4. LCD display

Display of liquid crystal is the shorthand for liquid crystal display. This kind of display module with electronics is used in many different circuits and gadgets, including computers, televisions, calculators, cell phones, and so forth. Seven segments and multi-segment light-emitting diodes are the most popular uses

for these displays. The primary benefits of using this component are its inexpensive cost, its programming simplicity, its videos, and the reality that it does not impose any limitations on the display of special characters, animations, or other content.

## 2.2. Architecture for software

It includes Mu editor and ThingSpeak.

### 2.2.1. Mu editor

Mu is a Python editor for beginners that aims to make the learning process more enjoyable. It gives students the opportunity to succeed right away, which is crucial when learning something new. Mu is a straightforward code editor for Adafruit CircuitPython boards. It runs on Windows, MacOS, Linux, and the Raspberry Pi and is developed in Python. The serial console is built-in, so you may obtain quick feedback from the serial output of your board.

### 2.2.2. ThingSpeak

An IoT analytics platform powered by the cloud called ThingSpeak is used to collect, visualize, and analyze real-time data streams. With ThingSpeak, we can communicate information generated by our devices, see real-time data visualization, and send alerts [13]. An open-source program called ThingSpeak, which runs on Ruby, enables users to interact with internet-connected devices. It makes access to information, retrieval, and logging easier by giving social network websites and devices an API. ThingSpeak was developed in 2010 by ioBridge as a tool to assist IoT developers. You may send data straight to ThingSpeak from any device that is connected to the internet via a Rest API or MQTT. The Things Network, Senet, the Libelium Meshlium gateway, and Particle.io cloud-to-cloud integrations enable sensor data to be delivered to ThingSpeak across LoRaWAN® and 4G/3G cellular networks.

## 3. Results and Discussions

This section involves the description about the following aspects.

- (1) Dataset
- (2) Predictor Model Description
- (3) Predictor Model Analysis Result
- (4) Hardware Implementation
- (5) Results obtained

### 3.1. Dataset

The dataset contains 303 persons' data. The dataset includes 8 input parameters which are Age, Gender, Blood Pressure, Blood sugar, Cholesterol level, genetic issue, heart rate, oxygen level saturation, and an output parameter. The dataset utilized is obtained from the people living in the southern region of Tamil Nadu. Analysis of the dataset is shown in Figures 3, 4, and 5. The characteristics of Figure 3 show the People having male gender marked as "1" are more than twice the People having female gender marked as "0." Figure 4 shows the number of people having Fasting Blood sugar in the dataset. Figure 4 shows bar chart of two classes "0" and "1," where "0" indicates the class of people who are having a Blood sugar value less than 120 mg dL and "1" indicates the class of people who are having a Blood sugar value more than 120 mg dL. Figure 5 shows the number of people already having chance of getting cardiac arrest by genetically. In this, "0" indicates a class of people who are not having genetic problem of this cardiac arrest and "1" indicates a class of people having genetic problem of this cardiac arrest.

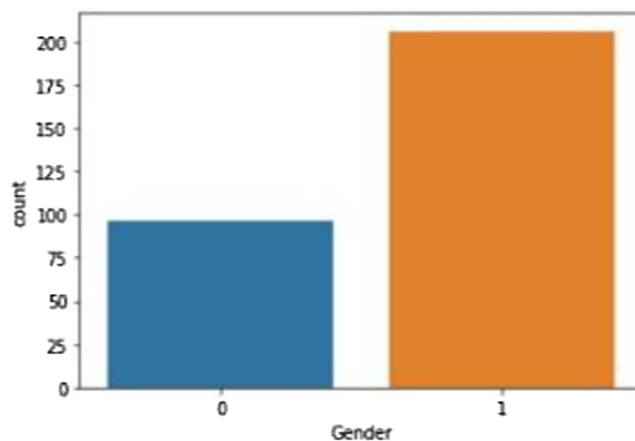


Figure 3. Dataset analysis result in terms of gender

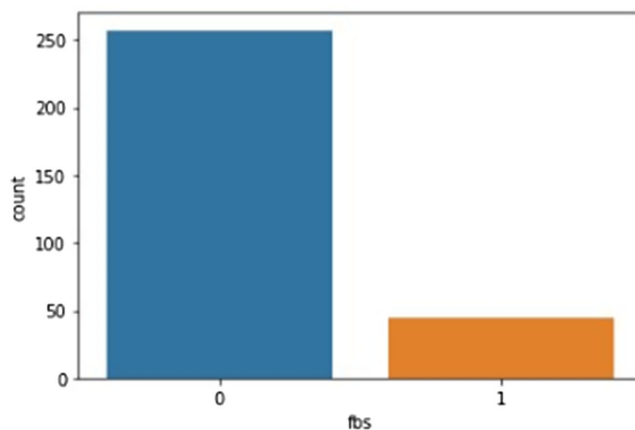


Figure 4. Dataset analysis result in terms of blood pressure

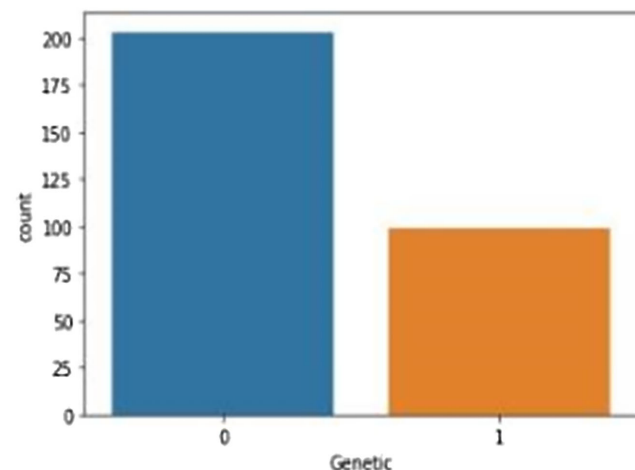


Figure 5. Dataset analysis result in terms of genetic issue

### 3.2. Predictor model

For developing real-time smart cardiac monitoring system for predicting the occurrence of heart attack, the application software algorithm is to be built. Application software should include a predictor model built with the help of dataset. The predictor

model can be built out of dataset using various software algorithm. To identify the best predictor model for cardiac detection, various artificial intelligence-based models are trained and tested with the above dataset. The accuracy for the predictor model is very important. By implementing different algorithms in the code, the algorithm having higher accuracy is selected and smart cardiac monitoring system was built out of it [14].

### 3.2.1. Machine learning algorithms

As the name suggests, machine learning is the process of learning without the requirement for direct human involvement or explicit programming. High-quality data are first supplied to them as part of this machine learning technique, and then, they are trained by building numerous machine learning models using different algorithms and the data. The kind of information we are dealing with and the process we are trying to automate the algorithms we apply [15]. The following is a list of some of the most popular machine learning algorithms.

#### (1) SVM (Support Vector Machine)

One of the most popular supervised learning techniques for categorization and problems with regression is SVM, or support vector machine. Finding the ideal line or conclusion boundary for classifying space with  $n$  dimensions into classes is the goal of the SVM method. This will enable future additions of data points to be easily added to the appropriate category. One term for the ideal choice boundary is a hyperplane. SVM is used to choose the extreme points and/or vectors that help build the hyperplane. The algorithm is referred to as a SVM [16], and support vectors are the worst-case scenarios.

#### (2) Bernoulli Naïve Bayes algorithm

This is applicable to discrete data and operates on the Bernoulli distribution. The main characteristic of Bernoulli Naive Bayes is that it only takes binary values for attributes like true or inaccurate, either positive or negative, whether it succeeded or failed, 0 or 1, and so forth. When the feature values are binary, we are aware that the Bernoulli Naive Bayes classifier must be used [17].

#### (3) Gaussian Naive Bayes

This Naive Bayes variant adheres to the Gaussian distribution's normality and accepts continuous data. One typical assumption made when handling continuous data is that each class's continuous values follow a normal or Gaussian, distribution. Gaussian Naive Bayes takes a somewhat different method and is effective. When dealing with continuous data, the Gaussian distribution is used [17].

#### (4) Logistic Regression

The linear regression algorithm identifies continuous values, whereas the algorithm logistic regression predicts discrete values. This suggests that for binary classification, logistic regression is a preferable alternative. In logistic regression, an event is categorized as 1 if it occurs and 0 if it does not. As a result, the chance of a specific event occurring is predicted using the predictor variables.

#### (5) Random Forest Algorithm

Using supervised learning methods, Random Forest is a widely recognized machine learning algorithm. It can be applied to both regression and classification problems in machine learning. Its foundation is ensemble learning, an approach that combines several

classifiers to tackle a challenging issue and improve the performance of the model. The Random Forest classifier enhances the projected accuracy of a dataset by averaging the outcomes of several decision trees on various dataset subsets. Accuracy increases and overfitting risk decreases with the number of trees in the forest.

### 3.3. Predictor model analysis result

Various predictor model discussed in Section V was analyzed on the defined dataset in Section IV, and the accuracy of those analyzed models is shown below as case-wise.

#### (1) Case 1

Out of total 61 random test samples, 47 samples (21 – True Heart attack scenario and 26 – True no heart attack scenario) were predicted correctly and remaining 14 (8 – False heart attack scenario and 6 – False no heart attack scenario) were not predicted correctly. The accuracy of SVM built using the discussed dataset is 77.05%.

#### (2) Case 2

Out of total 61 random test samples, 47 samples (22 – True Heart attack scenario and 25 – True no heart attack scenario) were predicted correctly and remaining 14 (9 – False heart attack scenario and 5 – False no heart attack scenario) were not predicted correctly. The accuracy of Bernoulli NB algorithm built using the discussed dataset is 77.05%.

#### (3) Case 3

Out of total 61 random test samples, 51 samples (22 – True Heart attack scenario and 29 – True no heart attack scenario) were predicted correctly and remaining 10 (5 – False heart attack scenario and 5 – False no heart attack scenario) were not predicted correctly. The accuracy of Gaussian NB algorithm built using the dataset is 83.61%.

#### (4) Case 4

Out of total 61 random test samples, 52 samples (24 – True Heart attack scenario and 28 – True no heart attack scenario) were predicted correctly and remaining 9 (6 – False heart attack scenario and 3 – False no heart attack scenario) were not predicted correctly. The accuracy of support logistic regression algorithm built using the discussed dataset is 85.25%.

#### (5) Case 5

Out of total 61 random test samples, 52 samples (24 – True Heart attack scenario and 28 – True no heart attack scenario) were predicted correctly and remaining 9 (6 – False heart attack scenario and 3 – False no heart attack scenario) were not predicted correctly. The accuracy of the algorithm random forest built using the discussed dataset is 85.25%.

Table 1 gives a clear picture of various predictor model built out of the defined dataset as discussed in section A. The Best predictor model is chosen to be Random Forest Regression for developing the smart cardiac monitoring and detection system in real time.

### 3.4. Hardware implementation

The MAX30100 sensor is used by the system to first gather health data. Subsequently, the Raspberry Pi 3 receives the sensor values. Following that, Raspberry Pi will use the system program

**Table 1. Accuracy and confusion matrix of algorithm with details of TP, TN, FP, and FN**

ALGORITHM	TRUE	TRUE	FALSE	FALSE	ACCURACY (%)
	POSITIVE (TP)	NEGATIVE (TN)	POSITIVE (FP)	NEGATIVE (FN)	
Random Forest Regression	24	28	6	3	85.25
Logistic Regression	24	28	6	3	85.25
Gaussian NB	22	29	5	5	83.61
Bernoulli NB	22	25	9	5	77.05
Support Vector Classifier (SVC)	21	26	8	6	77.05

that uses the predictor model to process the sensor data it has received [12]. The emergency room will automatically receive an alarm message if there is an unexpected rise in heart rate. After that, the entire procedure will begin anew. The following connections are made between the Raspberry Pi [18] and the MAX30100 sensor: The sensor’s Vin pin is attached to pin 4 (5V) of the Raspberry Pi, its GND pin is coupled to pin 6 (GND), its SCL pin is attached to pin 5 (GPIO 3) of the Raspberry Pi [19], and its SDA pin is attached to pin 3 (GPIO 2) of the Pi.

The MAX30100 sensor communicates with Raspberry pi through I2C Communication. The I2C address of the sensor connected to raspberry pi is 0x+57. To detect if the sensor is connected to Raspberry pi or not, the following command is used in the terminal.

```
sudo i2c detect -y 1
```

The sensor is connected to the Raspberry pi as the I2C address of the sensor is detected in the terminal by using the above command. The code for the reading the data from the sensor is written in python and is dumped in the Raspberry pi [20, 21]. The data for three persons are taken for the differentiation. When a person placed the finger on the sensor, the data are received and are displayed on the monitor.

### 3.5. Results obtained

The file rc.local is used to run a program on Raspberry pi [16] at start-up. Commands can be added to the rc.local file to have a command or application run when the Pi boots. This is very useful if Pi wants to be powered up without a monitor connected and have it run a program without any configuration or manual start. Edit the file /etc/rc.local on the terminal of the Pi. It must be edited with root privileges:

```
sudo nano /etc/rc.local
```

Add commands to run the Python application [22], ideally with absolute file location referring (complete file path are preferred). Make sure the line exit 0 is at the end of the file, then save and exit. To exit nano, use Ctrl-x and then Y. If the program runs indefinitely (infinite loop) or is unlikely to exit, it must be forked by including an ampersand (“&”) to the end of the instruction, such as:

```
sudo python /home/pi/Final.py &
```

After rebooting the Pi [23] by the command *sudo reboot*, the program we edited in rc.local starts to run automatically. The program is written in the way to run and display to enter the parameters in the LCD display and the values of some parameters are entered through the matrix keypad [17]. Likewise, the value for the other parameters like Age, Blood pressure, Blood sugar, cholesterol, and genetic disorder have to be entered through matrix keypad.

After entering the value to the genetic parameter, the heart rate and oxygen saturation level are measured by placing the finger tip on the MAX30100 heart rate sensor. After receiving the value of the input parameters, the algorithm predicts the status of the heart and displays

it in the LCD display. The result obtained shows whether the heart is in good condition along with a parameter value [24, 25]. Here, the output parameter value, if is less than 0.5 then the heart is in good condition and 0 is taken as field value then it is sent to the ThingSpeak cloud server through the API key and if the output parameter values is more than 0.5, prediction is vice versa. Field 1 chart of the ThingSpeak cloud server represents the output of cardiac arrest prediction. Here, if the value zero is received the server does not react as the heart is in good condition. If the value 1 is received, the server sends the alert message to the patient’s family members and to the hospital. The indication widget is created in the Freeboard platform. This pane indicates the output according to the value received from the ThingSpeak server which is either 1 or 0. By using ThingHTTP in ThingSpeak network, then the alert message is sent to the patient’s family members and hospital with the help of IFTTT application. If the field value is 1, the webhook receives a web request then it sends the e-mail to the respective people.

### 4. Conclusion

In this project, edge computing-based smart cardiac monitoring and detection is designed and implemented. It has provided with 85% accuracy in the prediction of a cardiac arrest. This proposed approach allows the early detection about the possibility of occurrence of one of cardiac-related problems, heart attack. The system’s primary purpose is to develop a low-cost, accurate gadget for monitoring heart rhythm and alerting patients. But this paper just presents about the small prototype designed as the initiation of this gadget development. This device continuously monitors the heart rate and oxygen saturation level of the patient along with the other input parametric data and alerts the appropriate family members and hospital in the event of an emergency. As a result, many lives can be spared, and cardiac arrest therapy can be provided in a relatively short time. By improving the dataset, the model will be more accurate and good for commercial as well as personal usage. In this project, MAX30100 sensor is used to detect heart rate and oxygen saturation level. In future, more sensors like temperature sensor to detect change in temperature level, pressure sensor to detect blood pressure level, and blood glucose sensor to detect blood sugar level can also be added along with heart rate sensor in this project’s prototype to predict the occurrence of the cardiac arrest in real time.

### Ethical Statement

This study does not contain any studies with human or animal subjects performed by any of the authors.

### Conflicts of Interest

The authors declare that they have no conflicts of interest to this work.



## Data Availability Statement

The data that support this work are available upon reasonable request to the corresponding author.

## References

- [1] Majumder, A. J., Elsaadany, M., Izaguirre, J. A., & Ucci, D. R. (2019). A real-time cardiac monitoring using a multisensory smart IoT system. In *IEEE 43rd Annual Computer Software and Applications Conference*, 281–287. <https://doi.org/10.1109/COMPSAC.2019.10220>
- [2] Bhardwaj, H., Bhatia, K., Jain, A., & Verma, N. (2021). IOT based health monitoring system. In *6th International Conference on Communication and Electronics Systems*, 1–6. <https://doi.org/10.1109/ICCES51350.2021.9489207>
- [3] Priya, A. D., & Sundar, S. (2019). Health monitoring system using IoT. In *International Conference on Vision Towards Emerging Trends in Communication and Networking*, 1–3. <https://doi.org/10.1109/ViTECoN.2019.8899434>
- [4] Khan, I., Zeb, K., Mahmood, A., Uddin, W., Khan, M. A., Islam, S., & Kim, H. J. (2019). Healthcare monitoring system and transforming monitored data into real time clinical feedback based on IoT using Raspberry Pi. In *2nd International Conference on Computing, Mathematics and Engineering Technologies*, 1–6. <https://doi.org/10.1109/ICOMET.2019.8673393>
- [5] Ahmid, M., Kazar, O., Benharzallah, S., Kahloul, L., & Merizig, A. (2020). An intelligent and secure health monitoring system based on agent. In *IEEE International Conference on Informatics, IoT, and Enabling Technologies*, 291–296. <https://doi.org/10.1109/ICIoT48696.2020.9089602>
- [6] Reshma, A. B. M. B., Rejee, W. H. N., Revathy, S., & Jabez, J. (2019). Recommendation of nearest heart care center during emergency using IoT. In *International Conference on Intelligent Computing and Control Systems*, 1056–1060. <https://doi.org/10.1109/ICCS45141.2019.9065409>
- [7] Ruman, M. R., Barua, A., Rahman, W., Jahan, K. R., Roni, M. J., & Rahman, M. F. (2020). IoT based emergency health monitoring system. In *International Conference on Industry 4.0 Technology*, 159–162. <https://doi.org/10.1109/I4Tech48345.2020.9102647>
- [8] Patil, M., Madankar, A., & Khandait, P. D. (2019). Heart rate monitoring system. In *5th International Conference on Advanced Computing & Communication Systems*, 574–576.
- [9] Sachdeva, P., & Katchii, S. (2014). A review paper on Raspberry Pi. *International Journal of Current Engineering and Technology*, 4(6), 3818–3819.
- [10] Kora, P., Rajani, A., Chinnaiah, M. C., Swaraja, K., & Meenakshi, K. (2021). Iot based wearable monitoring structure for detecting abnormal heart. In *International Conference on Sustainable Energy and Future Electric Transportation*, 1–4. <https://doi.org/10.1109/SeFet48154.2021.9375787>
- [11] Gupta, S., Dahiya, D., & Raj, G. (2018). Remote health monitoring system using IOT. In *International Conference on Advances in Computing and Communication Engineering*, 300–305. <https://doi.org/10.1109/ICACCE.2018.8441731>
- [12] Chaudhari, H. (2015). Raspberry Pi technology: A review. *International Journal of Innovative and Emerging Research in Engineering*, 2(3), 83–87. <https://www.semanticscholar.org/paper/Raspberry-Pi-Technology%3A-A-Review-Chaudhari-Gadge/7cd7dc930715342e38a82699be072264438daf97>
- [13] Yeri, V., & Shubhangi, D. C. (2020). IoT based real time health monitoring. In *Second International Conference on Inventive Research in Computing Applications*, 980–984. <https://doi.org/10.1109/ICIRCA48905.2020.9183194>
- [14] Tamilselvi, V., Sribalaji, S., Vigneshwaran, P., Vinu, P., & GeethaRamani, J. (2020). IoT based health monitoring system. In *6th International Conference on Advanced Computing and Communication Systems*, 386–389. <https://doi.org/10.1109/ICACCS48705.2020.9074192>
- [15] Chauhan, U., Kumar, V., Chauhan, V., Tiwary, S., & Kumar, A. (2019). Cardiac arrest prediction using machine learning algorithms. In *2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies*, 886–890. <https://doi.org/10.1109/ICICICT46008.2019.8993296>
- [16] McManus, S., & Cook, M. (2021). *Raspberry Pi for dummies*. USA: Wiley.
- [17] Kakria, P., Tripathi, N. K., & Kitipawang, P. (2015). A real-time health monitoring system for remote cardiac patients using smartphone and wearable sensors. *International Journal of Telemedicine and Applications*, 2015, 373474. <https://doi.org/10.1155/2015/373474>
- [18] Moore, J. F. (2017). *Raspberry Pi boot process*. Retrieved from: <https://www.lions-wing.net/maker/raspberry-1/boot.html>
- [19] Halfacree, G. (2018). *The official Raspberry Pi beginner's guide: How to use your new computer*. UK: Raspberry Pi Press.
- [20] Fanucci, L., Saponara, S., Bacchillone, T., Donati, M., Barba, P., Sánchez-Tato, I., & Carmona, C. (2013). Sensing devices and sensor signal processing for remote monitoring of vital signs in CHF patients. *IEEE Transactions on Instrumentation and Measurement*, 62(3), 553–569. <https://doi.org/10.1109/TIM.2012.2218681>
- [21] Jain, P., Rusin, C., Puri, K., Patel, R., Gagne, V., & Valdes, S. (2019). 247: A novel real-time patient monitoring system for detection of arrhythmias in congenital heart disease. *Critical Care Medicine*, 47(1), 105. <https://doi.org/10.1097/01.ccm.0000551001.36494.32>
- [22] van Rossum, G. (1995). *Python tutorial. Technical report CS-R9526*. Retrieved from: <https://gvanrossum.github.io/Publications.html>
- [23] Ambrosino, N., Vitacca, M., Dreher, M., Isetta, V., Montserrat, J. M., Tonia, T., . . . , & Vaghegghini, G. (2016). Tele-monitoring of ventilator-dependent patients: A European Respiratory Society Statement. *European Respiratory Journal*, 48(3), 648–663.
- [24] Dennis, A. K. (2016). *Raspberry Pi computer architecture essentials*. UK: Packt Publishing.
- [25] Kölling, M. (2016). Educational programming on the Raspberry Pi. *Electronics*, 5(3), 33. <https://doi.org/10.3390/electronics5030033>

**How to Cite:** Murugesan, S., & Naganathan, A. (2024). Edge Computing-Based Cardiac Monitoring and Detection System in the Internet of Medical Things. *Medinformatics*. <https://doi.org/10.47852/bonviewMEDIN42022430>