# SECURING A PATIENT'S HEALTH REMOTELY USING LIGHT CRYPTOGRAPHY

ELAD Georgette Jocelyne(Corresponding author)
Computer and Automatic Engineering Laboratory, Higher Normal School of Technical Education of the University of Douala
Douala, Cameroun
georgetteelad@gmail.com

NNEME NNEME Leandre
Computer and Automatic Engineering Laboratory, Higher Normal School of Technical Education of the University of Douala
Douala, Cameroun
leandren@gmail.com

MENGATA MENGOUNOU Ghislain
Computer and Automatic Engineering Laboratory, Higher Normal School of Technical Education of the University of Douala
Douala, Cameroun
mengounal@yahoo.fr

**Abstract**

**Light cryptography is a variant of conventional cryptography designed to minimize resource requirements. It is particularly well suited to the connected devices of the Internet of Things (IoT). Remote patient monitoring using the Internet of Things is a fast-moving field. This article presents a secure system for remotely monitoring vital patient parameters such as body temperature, heart rate and rhythm, while geolocating the patient. The device is placed on the patient to record his or her parameters, as well as geographical data, and then sends it to the doctor. Remote management of health data, such as vital parameters and patient geolocation, requires enhanced security due to the sensitivity of the information transmitted to healthcare professionals. Wireless communication networks are particularly vulnerable to security threats. This is why cryptography plays a crucial role in securing the data transmitted, from the sensor to the main node, before it is displayed online. In this context, the Ngrock platform is used to create a secure tunnel, remote local access to the module. A study of the most suitable and fastest algorithms was carried out, and we opted for the AES algorithm, recognized as one of the best algorithms thanks to its simplicity, lightness and suitability for IoT applications. At the end of the project, we produced a 3D-modelled prototype in line with the initial specifications, accompanied by an AES algorithm from a dedicated library, to ensure data security. Performance analysis showed that the algorithm used had no noticeable impact on system execution time, with an observed lag of around 1ms.**

**Keywords : IoT; temperature; cardiac frequency; heartbeat; geolocation; cryptography**

## 1. INTRODUCTION

In today's digital age, the integration of the Internet has transformed various sectors, including healthcare[1] . The term Internet of Things (IoT[2], [3]was coined by Kevin Ashton during a presentation at Proctor & Gamble in 1999. He is one of the founders of the Automatic Recognition Laboratory at the Massachusetts Institute of Technology[4] . The emergence of Internet of Things (IoT) technology in healthcare; monitoring systems have revolutionized the way we approach healthcare[1], [5] . With the widespread use of the Internet, coupled with the increasing efficiency of devices and gadgets, we can now monitor patients 24 hours a day thanks to the Internet of Things (IoT)-based health monitoring systems[1] . Healthcare monitoring tasks have now become a user-friendly and persistent environment compared to the old traditional clinical environment; The variety of monitoring purposes has widened considerably, ranging from patents to intensive care, such as an ambulance patient or one suffering from chronic diseases[6] . Sensors can therefore be installed on parts of the patient's body to measure vital signs such as temperature, heart rate, etc., or even to locate the patient in question via a GPS (Global Positioning System) module, in order to present information to the patient or send the information to the doctor monitoring the patient remotely. The architecture describing the Internet of Things (IoT) is shown in Fig. 1 :

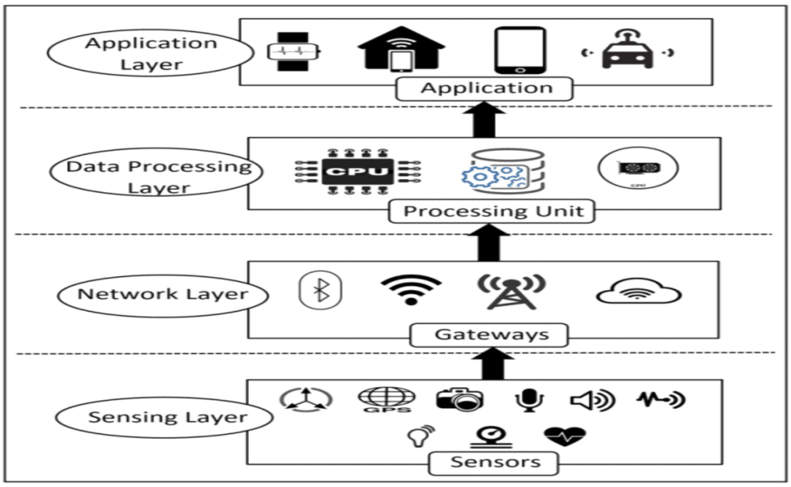


Fig.1. IoT Architecture [7]

The architecture shown in Fig.1. Features sensors which, via a Wifi or Bluetooth network, communicate or send data to a central node, which in turn sends data to web or mobile applications.

Security and protection in the Internet of Things (IoT) have remained a real concern due to the heterogeneous idea of the gadgets' enormous reach and weaknesses in working environments[8] . IoT devices often collect and transmit sensitive data, such as personal information, medical data or financial information. The security of this data is essential to protect users' privacy and prevent malicious attacks. Cryptography is proving to be a promising solution to data breach problems, with the implementation of conventional cryptography on resource-limited sensors compounding the security challenges[9] . Cryptography[10], [11] is defined as the science of secrecy, or the science of transforming information into a non-readable format to protect it from unauthorized access. Data confidentiality can be ensured using both asymmetric and symmetric encryption[12] . Symmetric-key cryptography is also known as secret-key or shared-key cryptography. In this type of mechanism, the sender and receiver share a common key for encryption and decryption[13] . The method follows the self-certification method, i.e. the key is self-certified. The key must be shared via secret communication. If compromised, the encrypted message can be easily decrypted by the attacker. This type of cryptographic technique is necessary because it provides a faster service without using many resources. Various algorithms have been developed to date to describe symmetric key cryptography. We have AES, DES, 3DES, Blowfish . [13]The principle of asymmetric cryptography is shown in Fig. 2 :

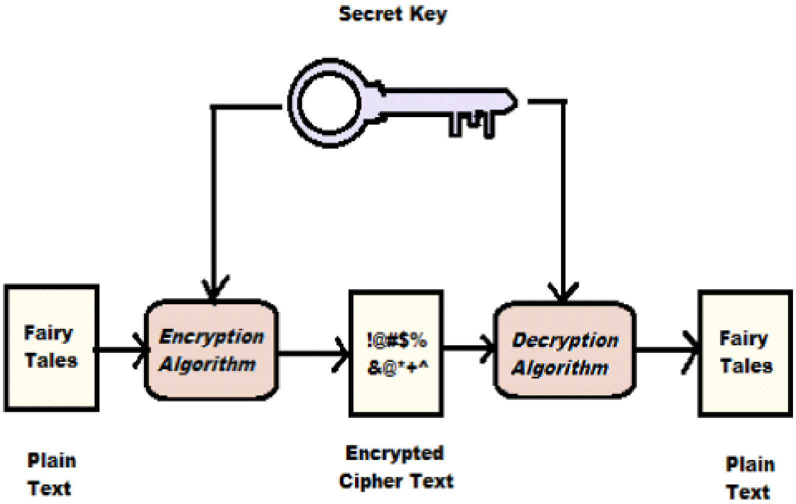


Fig. 2. Symmetric key cryptography[13]

Asymmetric key cryptography is also known as public key cryptography. In this technique, the sender uses the recipient's public key for encryption, and the recipient uses his or her private key to decrypt the message. The concept of self-certification is absent in this context, but digital signatures are used to certify the keys. This method is more practical and offers enhanced authentication, while guaranteeing data confidentiality. Various algorithms are available to implement this encryption mechanism. These include RSA, Diffie-Hellman, ECC and Digital Signature Algorithm[13] . The principle of asymmetric cryptography is illustrated in Fig. 3 :

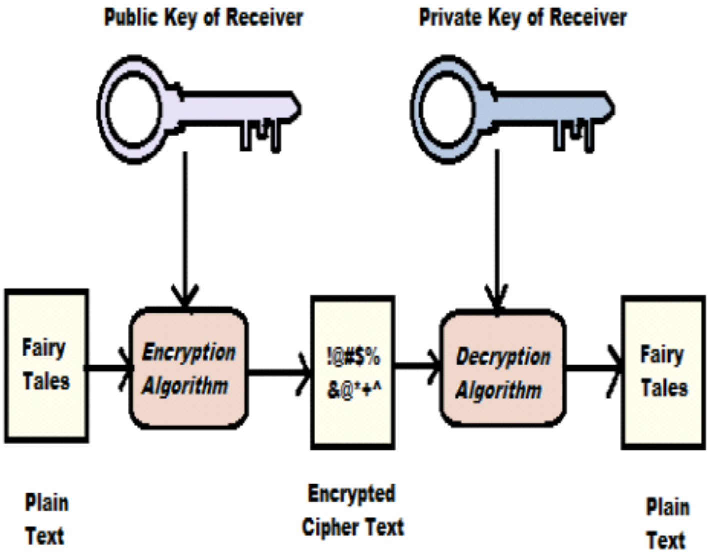


Fig. 3. Asymmetric key cryptography[13]

Cryptography plays a central role in this article, which is structured in five parts: the first part presents related work; the second part describes the functioning of the system; the third part proposes a comparative study of cryptographic algorithms and justifies the choice of the algorithm selected; the fourth part details the methods and tools used, as well as the realization of the system; and finally the fifth part presents the results obtained, associated with tests and discussions.

## 2. Ancillary Works

In this article, we will present a secure system for collecting parameters from a customer, as well as a secure medical monitoring system. To this end, we will present existing health monitoring systems .

### *2.1. health monitoring*

The classic method of patient health monitoring has been a staple of healthcare facilities such as hospitals and clinics since its inception in the early 2000s. Despite its extensive use over the years, this approach fails to accurately monitor a large number of patients within a defined timeframe. Healthcare staff, consisting of doctors and nurses, are responsible for physically examining and recording patient data. Patients must queue or wait their turn for examinations, while those confined to bed must be monitored 24 hours a day. Patient data is meticulously recorded by hand in a patient booklet and stored in physical form on shelves .[1]

### *2.2. Real-time health monitoring systems*

In[14] Amna Abdullah, Asma Ismael, Aisha Rashid, Ali Abou-ElNour, and Mohammed Tarique present a reliable patient monitoring system for professionals to monitor their patients hospitalized or living their normal daily life activities. In this work, we present a wireless healthcare monitoring system based on a mobile device that can provide real-time online information on a patient's physiological conditions. In addition, the proposed system is capable of sending an alarming message about the patient's critical health data via SMS or e-mail. Using the information contained in the text or e-mail message, the healthcare professional can provide advice.

## 3. Health monitoring via the Internet of Things (IoT)

In[15] Mohd Amirul Asyraf Razali, Murizah Kassim, Norakmar Arbain Sulaiman and Shuria Saaidin present the development of a real-time room condition monitoring system with ThingSpeak IoT with temperature and humidity measurement for room condition. Many conditions such as temperature and humidity monitoring systems have been designed previously, but some missing systems are identified when they don't provide adaptive connections and alerts to web pages on logging data collection.

In[16] Vishal Dineshkumar Soni presents the IoT as a means that can make a difference to patients' lives through its devices that can capture and monitor patient data and enable providers to obtain information without bringing patients in for a visit. The procedure can deliver patient results while avoiding potential communications for the risky process. However, the lack of an electronic health record (EHR) system. Integration is one of the major problems encountered when using IoT in the healthcare sector.

## 4. Description Of System Operation

It all starts with a patient's need to be monitored at a distance. These patients may be suffering from a disease such as heart failure or high blood pressure, and necessarily need regular monitoring of their heart rate, or they may want to be controlled. Patient monitoring involves taking parameters such as heart rate, temperature, etc. The system works as follows:

- Placing the device on the patient : The device will first have to be fixed to the patient's body; with a 3D-printed wristband, fitted with an integrated case to make it a unique component. The wristband will feature a screen and an electrocardiogram linked to the case. It will be positioned on the abdomen to optimize the collection of parameters, notably body temperature and heart rate
- Launch the system and share it on Ngrok : This stage involves actually launching the system and publishing the parameters retrieved from the local network online, giving doctors a real-time view of the patient's parameters. Encryption is used to exchange data between different sensors, and geolocation via a GPS module pinpoints the patient's position in the event of illness.

The operation of the system is described in Fig. 4 :

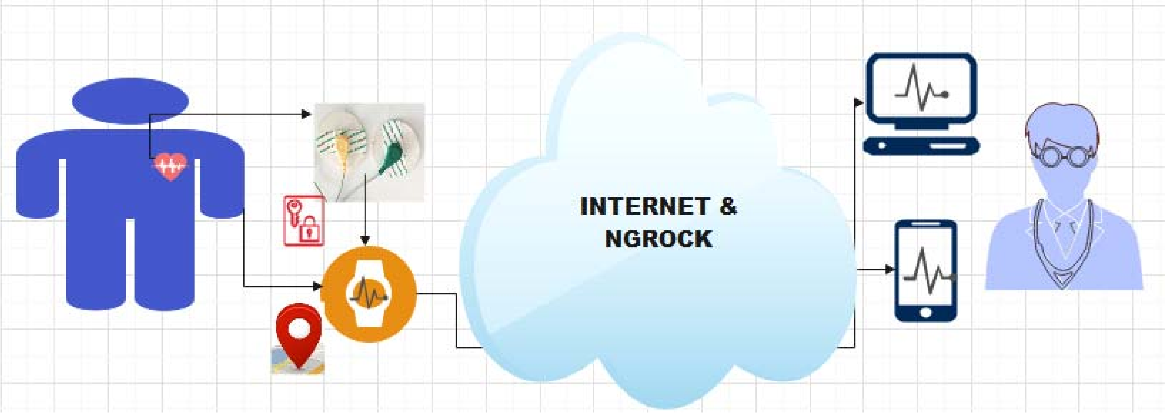


Fig. 4 : Description of system operation

## 5. Study Of Cryptographic Algorithms And Algorithm Selection

Our research is based on those carried out by Dutta et al in[17] those carried out by Shah et al in[7] . These reviews present cryptographic algorithms used par excellence in the Internet of Things (IoT). Table 1 shows the fundamental differences between symmetric and asymmetric key algorithms.

| | Cryptographic methods | |
|---|---|---|
| | Symmetric key cryptography | Asymmetric key cryptography |
| Key | A shared private key | A unique public and private key pair |
| Speed and complexity | Faster | Less fast due to different keys used |
| Number of keys | Proportional to number of users | Exponentially proportional to the number of users |
| Example | RSA, DSA, ECC | Stream encryption: Trivium,<br>Chacha, WG-8, Espresso, Grain 128<br>Figures by block: AES, DES,<br>3DES, Blowfish, Twofish |

Table 1 : Fundamental differences between symmetric and asymmetric key algorithms

Although block ciphers have low latency, they are the most sought-after and modified solutions for IoT security [8]. There are different types of block ciphers, such as Advanced Encryption Standard (AES), Data Encryption Standard (DES), 3DES, Blowfish, Twofish. Different approaches have been adopted by researchers to make these block ciphers lightweight and suitable for the IoT. Some other lightweight block ciphers currently being researched are Curupira, PRESENT, KATAN, LED, RECTANGLE... Table 2. shows a basic comparison between some common block ciphers[18], [19], [20] :

| Block Cipher | Key Size (bit) | Block Size (bit) | N° of Rounds | Characteristics |
|---|---|---|---|---|
| AES | 128, 192, 256 | 128 | 10, 12 , 14 | Excellent security, Flexible |
| FROM | 64 | 64 | 16 | Not very secure but flexible |
| 3DES | 112, 118 | 64 | 48 | Good security, flexible |
| Blowfish | 32, 448 | 64 | 16 | Excellent security, flexible |
| Twofish | 128, 192, 256 | 128 | 16 | Can't be broken remotely |
| Curupira | 96, 144, 192 | 96 | 96,144,192 | Less space required to store S-boxes |
| PRESENT | 80, 128 | 128 | 32 | Less gate count, less memory, Used for encrypting small amount of data |
| KATAN | 80 | 32, 48, 64 | 256 | Hardware oriented block cipher, inefficient software implementation, consumes too much energy, low throughput |
| LED | 64, 128 | 64, 128 | | Efficient hardware implementation, used for transmission of RFID tags |
| RECTANGLE | 80 | 64 | 25 | Hardware friendly, faster, gives high throughput |

Table 2. Basic comparison of block ciphers

The AES block size is 128 bits, while keys can be 128, 192 or 256 bits long. The AES algorithm has proven to be the most efficient of those presented in the table above. With a block length and key size of 128 bits, AES offers a high level of security while maintaining processing efficiency. To date, no shortcut attacks have been identified for versions reduced by more than 6 bits. These features make AES a reliable and robust choice for protecting sensitive data. We have therefore opted for the AES algorithm.

### *5.1. AES Algorithm*

Lightweight AES for the IoT has so far been a demanding research topic. An energy-efficient AES core has been presented by[21] . AES ,according to studies carried out in[22] presents an AES microarchitecture with a 32-bit data path for low power consumption targeting IoT applications. AES has been demonstrated in[23] to be lightweight and sufficiently suitable for IoT applications.

## 6. THE COMPONENTS

- ESP32-WROOM

Espressif's ESP32 WROOM is an affordable, energy-efficient microcontroller chip. It features Wi-Fi and Bluetooth capabilities, and is based on a highly integrated dual-core Tensilica Xtensa LX6 processor [24]. This element is presented in Fig.5 :

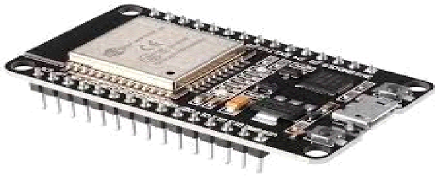


Fig. 5. ESP32-wroom chip[25]

- NEO-7M module

The NEO-7M module gives us the exact position of the patient at all times. This element is presented in Fig.6 :

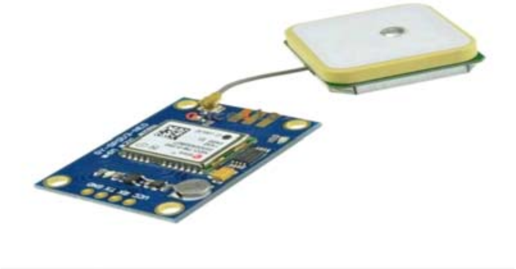


Fig. 6. NEO-7M*[26]*

- DS18B20 temperature sensor

The DS18B20 digital temperature sensor provides 9- to 12-bit (configurable) patient temperature measurements, offering variable accuracy. The one-wire DS18B20 interface enables simple, efficient communication with the ESP32 [27] , , .[28][29]. Fig. 7 shows a DS18B20 temperature sensor：

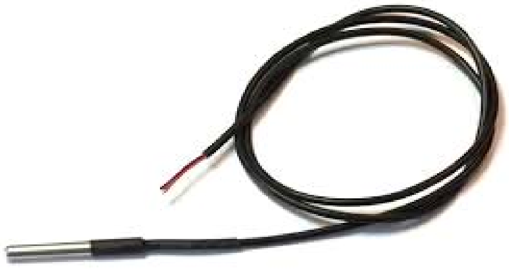


Fig. 7. *DS18B20[30]*

- HW-827

The pulse sensor provides a digital measurement of heart rate [31].This element is presented in Fig.8：

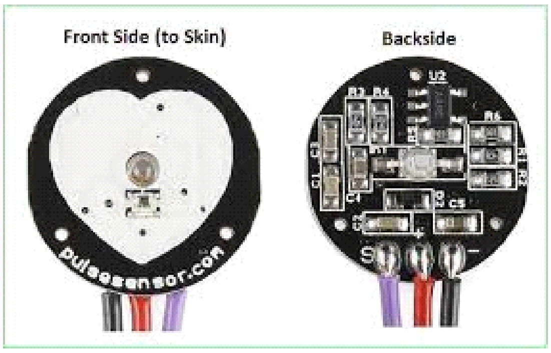


Fig. 8. pulse sensor*[32]*

- AD8232

The AD8232 sensor captures and amplifies the electrical signals emitted by the heart , , .[33][34][35]. Fig.9 shows a DS18B20 temperature sensor :

Fig. 9 .ECG*[36]*

- Oled screen

The organic light-emitting diode (OLED) is distinguished by its nature as an area light source, its low operating voltage and its non-toxic composition, and is particularly interesting for many applications .[37] Fig. 10 shows a OLED screen：

Fig. 10. oled display[38]

- Laptop computer

A laptop computer equipped with a set of specific pre-configured software tools for data acquisition on the web page. This element is presented in Fig.11 ：

Fig. 11. PC/Laptop

- Circuit : We first drew the circuit with the Fritzing software presented in Fig.12 :

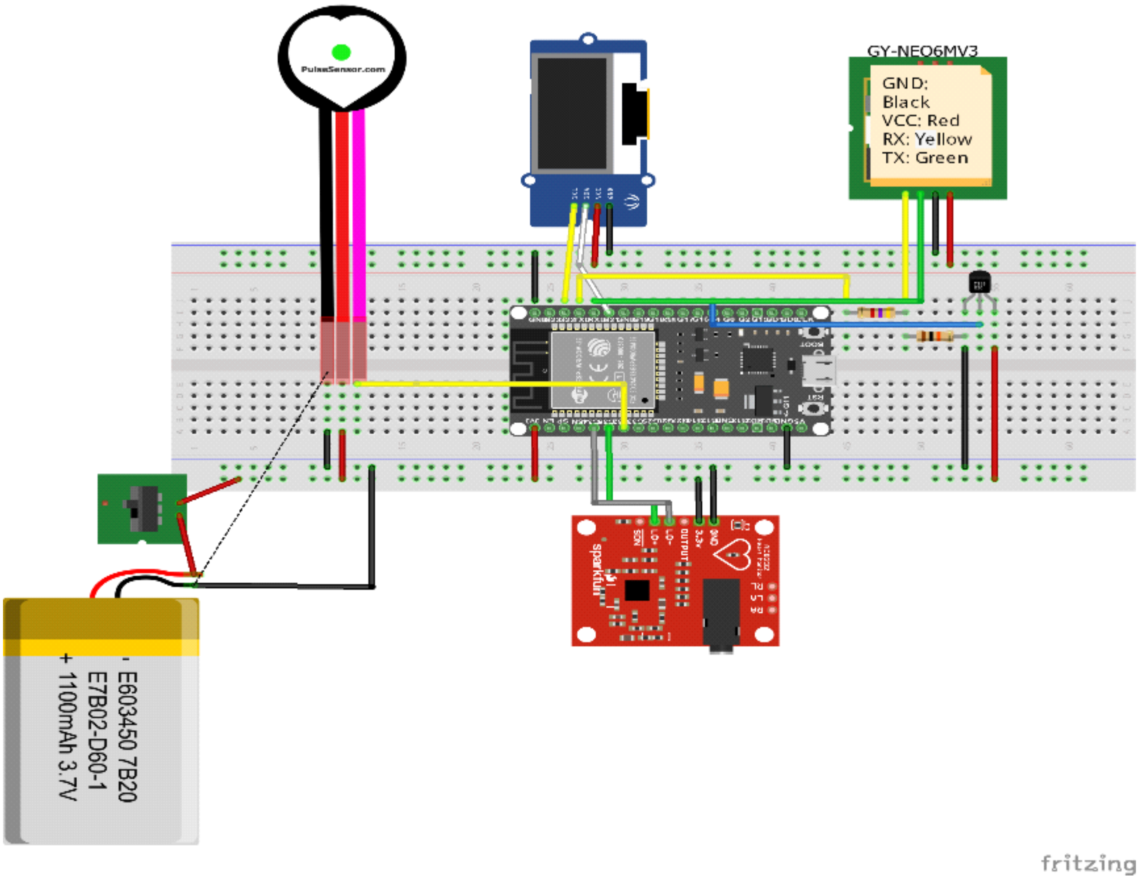


Fig. 12. Fritzing circuit

## 7. RESULTS AND PERFORMANCE

### *7.1. Results*

After modeling the circuit on fritzing, we proceeded to assemble the system. Fig. 13 shows the complete system before 3D modeling :

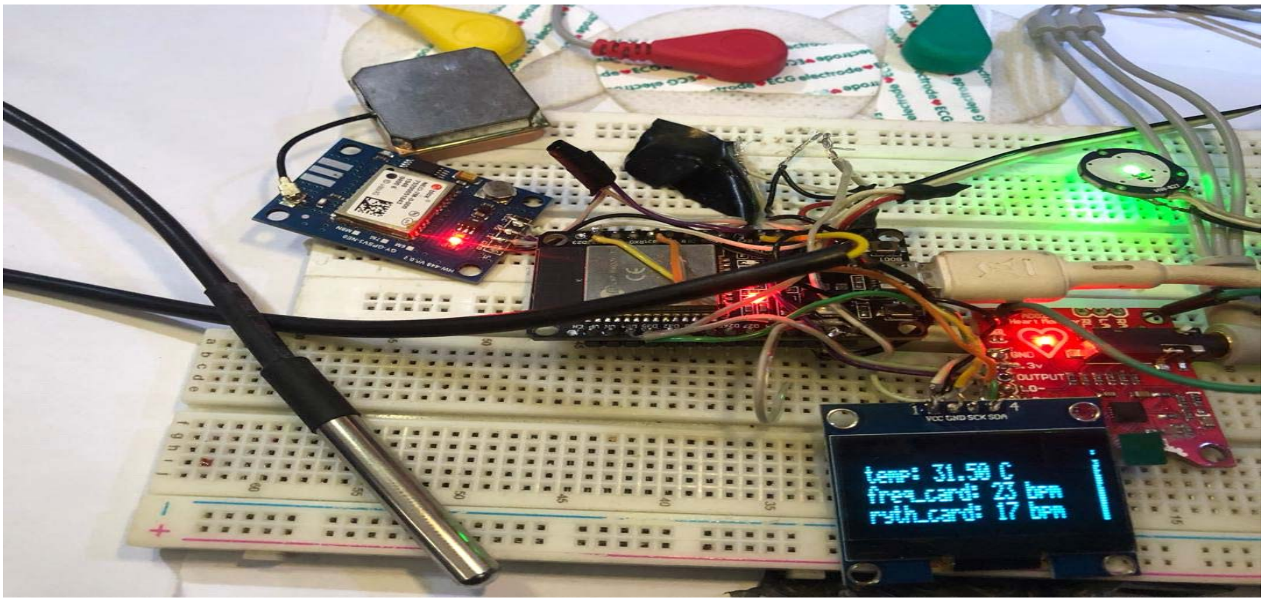


Fig. 13. System assembly

Fig. 14 shows the 3D model of the system :

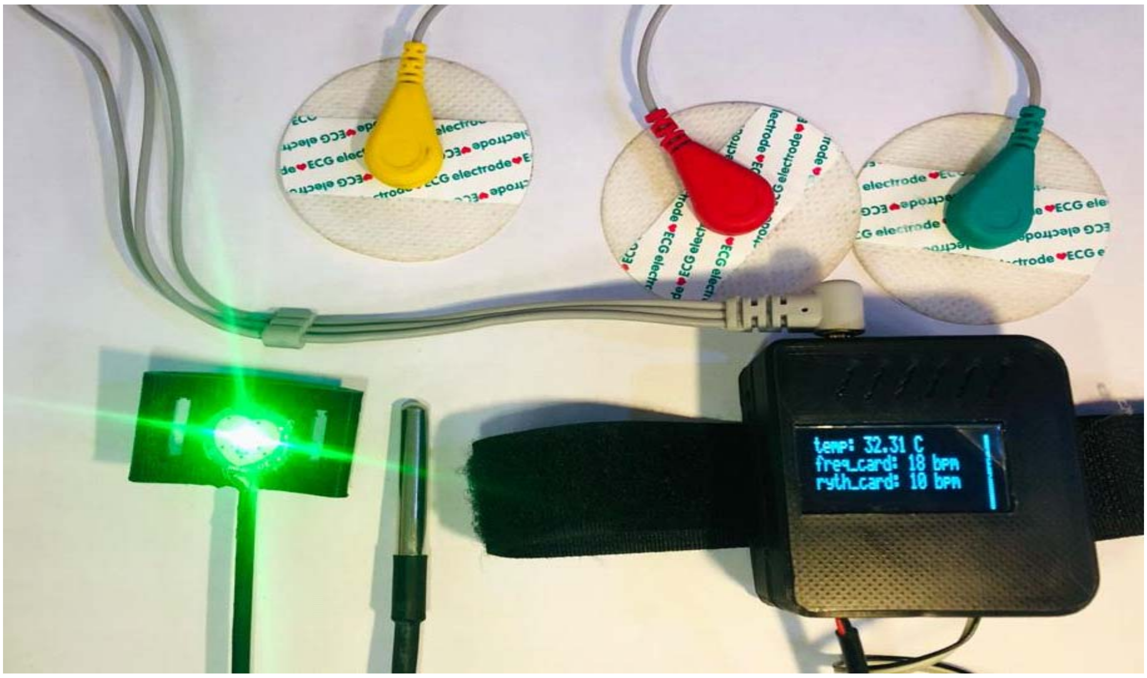


Fig. 14. 3D modeling

The 3D prototype was implanted on a patient and connected to the peripheral systems as shown in Fig.15 :

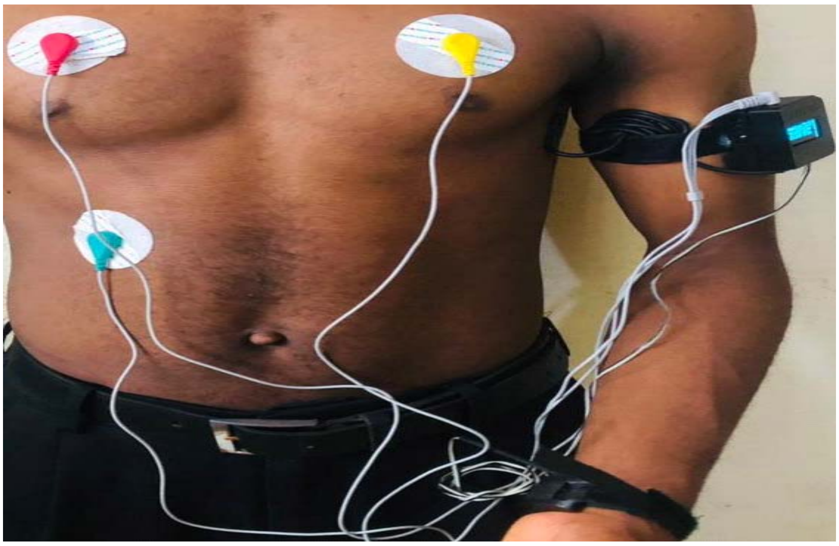


Fig. 15. System connected to the patient

The data collected was transmitted to a doctor via the ngrock platform, giving him access to the patient's vital parameters and geographical position. This is represented by Fig. 16 and Fig 17 :

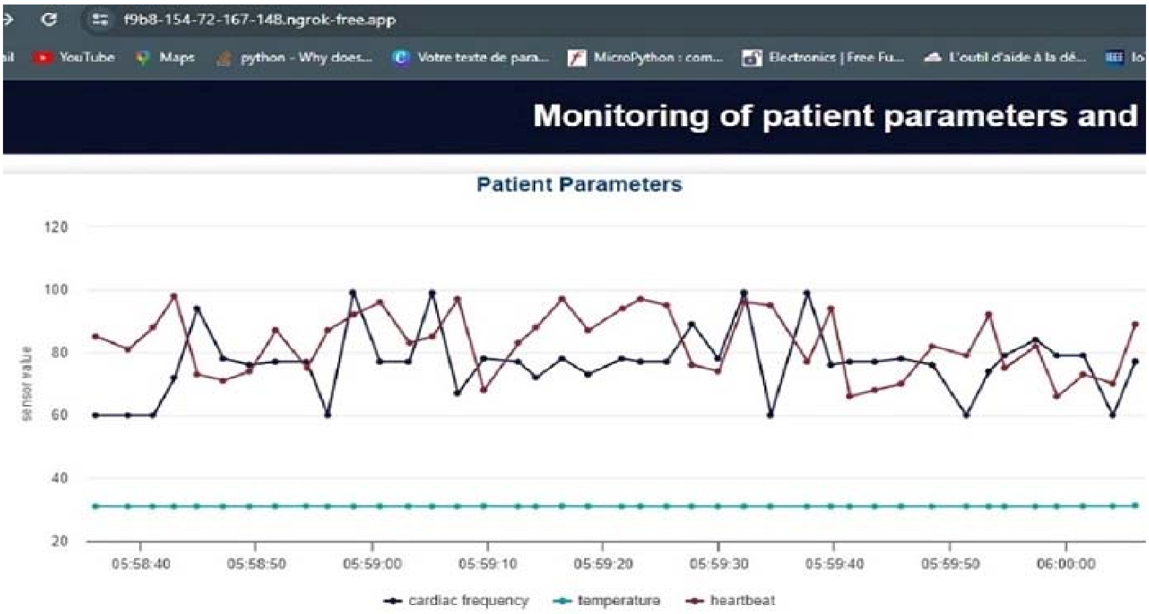


Fig.16. Monitoring of patient parameters in ngrock

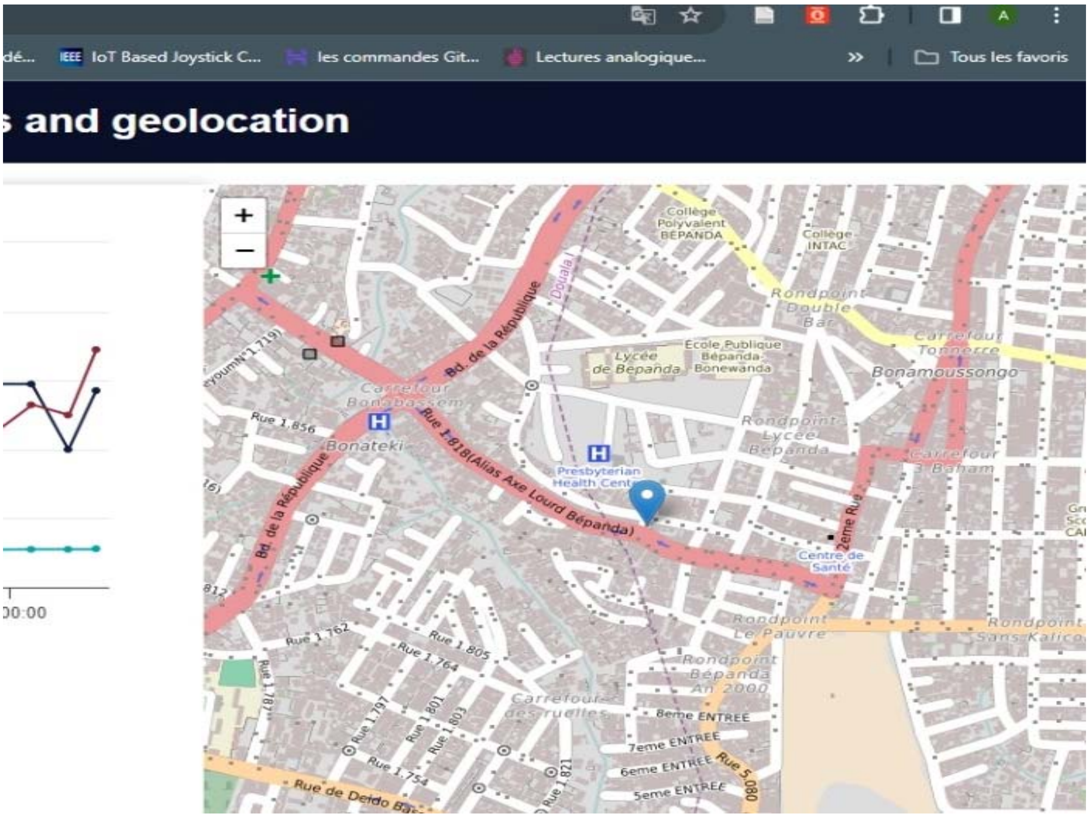


Fig.17. Patient geolocation in ngrok

The main functions which allowed us to have previous results are represented by Fig.18 :

```
String getSensorReadings(){
readings["sensor1"] = String(readingpulse());
  readings["sensor2"] = String(sonde());
  readings["sensor3"] = String(readECG());
  String jsonString = JSON.stringify(readings);
  return jsonString;
}
```

Fig.18. Retrieving sensor values

Fig.18 presents the function which allows the different values of the sensors to be retrieved. These values will subsequently be encrypted and decrypted end-to-end by cryptography functions. [39], [40] are documentations used for the functions that allowed us to encrypt and decrypt the data. The encryption function is shown in Fig. 19 :

```
String encrypt(String value_to_encrypt){
      Serial.println("\nCiphered text:");
       String text = cipher->encryptString(value_to_encrypt);
return text;
```

Fig.19. Encryption function

The decryption function is shown in Fig. 20 :

```
String decrypt(String Encrypt text)
      {
      //function to decrypt sensor values mSpiffs.writeFile(SPIFFS, "/test.txt",
textCrypter);
      String decryptValue=cipher->  decryptString(mSpiffs.getFile(SPIFFS,
"/test.txt"));
       return decryptValue;
    }
```

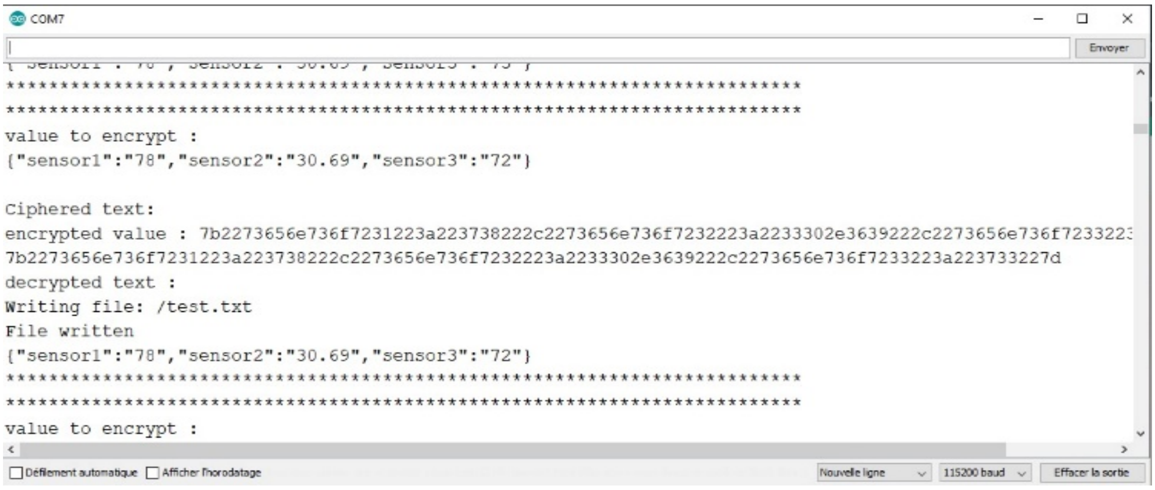
Fig. 20. Decryption Function

The call of these different functions to secure the values of the different sensors is presented in Fig. 21 :

```
String textCode = encrypt(getSensorReadings());
          //value of the sensors (getSensorReadings()) as parameter in the function
          //responsible for encrypting values
          Serial.println(textCode);
String decryptValue = decrypt(textCode); // encrypted value as parameter in the function
     // responsible for decrypting the encrypted value
          Serial.println(decryptValue);
```

Fig. 21. Encryption and decryption of sensor data

We took an example of data on the different sensors and displayed the end-to-end cryptogram finally decrypt and display the starting information, in Fig. 22 :



```
{"sensor1":"78","sensor2":"30.69","sensor3":"73"}
**************************************************************************
**************************************************************************
value to encrypt :
{"sensor1":"78","sensor2":"30.69","sensor3":"72"}

Ciphered text:
encrypted value : 7b2273656e736f7231223a223738222c2273656e736f7232223a2233302e3639222c2273656e736f7233223
7b2273656e736f7231223a223738222c2273656e736f7232223a2233302e3639222c2273656e736f7233223a223733227d
decrypted text :
Writing file: /test.txt
File written
{"sensor1":"78","sensor2":"30.69","sensor3":"72"}
**************************************************************************
**************************************************************************
value to encrypt :
```

Fig. 22. Cryptogram

## 8. Performance

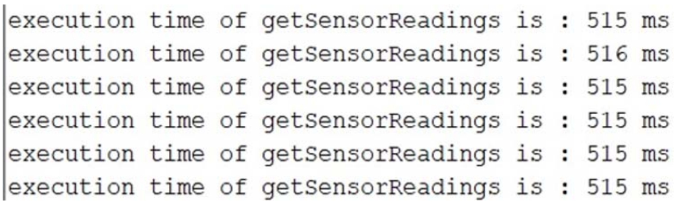
We estimated the performance of our system element by element. These performances studied are the frequency, the power, the flow and the surface area of each element. The performances are presented in Table 3 :

| Performance | | | | |
|---|---|---|---|---|
| Element | Frequency | Power | Speed | Area |
| PulseSensor | 433MHz | 10mW | 9600 -115200 bits/s | 47 mm x 25 mm x 13 mm |
| NEO-7 M GPS | 1 Hz | 28dBm | 1152000 bits/s | 20.3mm x 20.3mm x 7.1mm |
| DS18B20 | 600 Hz et 15 kHz. | / | 16 bits/s | 5 mm x 3 mm x 2,5 mm. |
| AD8232 | AD8232 | 50 mW | | 4 mm x 4 mm x 0.9mm |
| ESP32 | 260 MHz | 100 mW | In Wi-Fi is 150 Mbps<br>In Bluetooth is 1 Mbps | ESP32 |

Tableau 3. System Performance

We subsequently carried out a test as part of our work by estimating the transmission time of data from the sensors to the display using the encryption and decryption function on the one hand and on the other hand without the use of cryptography. , the tests were carried out and we received feedback from the following consoles :

- Transmission time tests without cryptography function is represented by Fig. 23 :

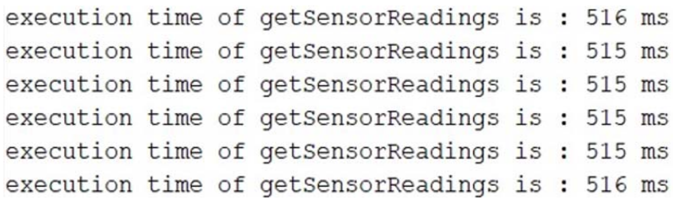


```
execution time of getSensorReadings is : 515 ms
execution time of getSensorReadings is : 516 ms
execution time of getSensorReadings is : 515 ms
execution time of getSensorReadings is : 515 ms
execution time of getSensorReadings is : 515 ms
execution time of getSensorReadings is : 515 ms
```

Fig. 23. execution time without cryptography function

We observe relative stability in the execution of the function in question, a variation is made of approximately 1ms but the rest of the execution is very stable, i.e. 515 ms.

- The transmission time tests with encryption and decryption functions are presented in the Fig. 24 :



```
execution time of getSensorReadings is : 516 ms
execution time of getSensorReadings is : 515 ms
execution time of getSensorReadings is : 515 ms
execution time of getSensorReadings is : 515 ms
execution time of getSensorReadings is : 515 ms
execution time of getSensorReadings is : 516 ms
```

Fig. 24. runtime with cryptography functions

We observe a fairly frequent variation using the cryptography library, a variation of 1ms, the execution time therefore varies between 515 and 516ms presenting a very positive impact of the algorithm on our system because the execution time is essentially the same. We have therefore represented a small graph materializing the execution trace in Fig. 25 :

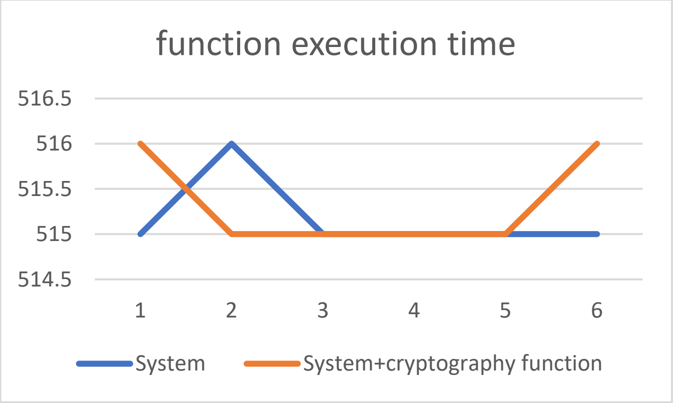


Fig.25. Variation in function execution time

We clearly notice that there are not really big differences between the execution time with cryptography function and without cryptography function. We can therefore conclude by saying that the cryptography library used did not have a real impact on the sensor network and will not slow down the transmission of information. We can talk about lightweight cryptography in this case

## CONCLUSION

Ultimately, our aim was to present an article aimed at remotely monitoring a patient by collecting his or her health parameters. These vital parameters, together with the patient's geolocation data, will be transmitted to the attending physician. The security of communications between the sensors and the heart of the system has been reinforced by AES algorithm, implemented using a dedicated library. To carry out this work, we have divided our document into four parts. The first part presents the overall operation of the system, illustrated by a block diagram; the second part, entitled study and choice of cryptographic algorithm, is devoted to the methodical selection of the optimum algorithm for guaranteeing data security in our system; the third part, entitled method and tools, details the methodological approach adopted and the set of tools used in designing the system; the final part, entitled results and discussions, presents an analysis of the results obtained during the testing phase of the final prototype. Once the system had been built, the test results confirmed the effectiveness of the chosen encryption algorithm, which ensures optimum data protection while preserving system performance. We noted a very high performance gain, with a maximum difference of **1ms**, which is more than enough to confirm the lightweight nature of our cryptography. At the end of our work, our aim is to develop a complete system for the prediction and detection of diseases such as heart failure, through a much more advanced solution.

## References

[1] J. A. J. Alsayaydeh, M. F. B. Yusof, M. Z. B. A. Halim, M. N. S. Zainudin, and S. G. Herawan, "Patient Health Monitoring System Development using ESP8266 and Arduino with IoT Platform," IJACSA, vol. 14, n° 4, 2023, doi: 10.14569/IJACSA.2023.0140467.

[2] H. Garg and M. Dave, "Securing IoT Devices and SecurelyConnecting the Dots Using REST API and Middleware," in 2019 4th International Conference on Internet of Things: Smart Innovation and Usages (IoT-SIU), Apr. 2019, pp. 1-6. doi: 10.1109/IoT-SIU.2019.8777334.

[3] T.-A. Tsai and F. J. Lin, "Enabling IoT Network Slicing with Network Function Virtualization", Advances in Internet of Things, vol. 10, n° 3, Art. n° 3, Jul. 2020, doi: 10.4236/ait.2020.103003.

[4] R. A. Mouha, "Internet of Things (IoT)", Journal of Data Analysis and Information Processing, vol. 9, n° 2, Art. n° 2, March 2021, doi: 10.4236/jdaip.2021.92006.

[5] N. Kulkarni et al, "Healthcare Monitoring System Using IoT", IJRASET, vol. 10, n° 12, pp. 341-345, Dec. 2022, doi: 10.22214/ijraset.2022.47819.

[6] M. Ajmal and N. Ahmed, "Privacy and Security Mechanisms for eHealth Monitoring Systems," ijacsa, vol. 8, n° 4, 2017, doi: 10.14569/IJACSA.2017.080470.

[7] P. Shah, M. Arora, and K. Adhvaryu, "Lightweight Cryptography Algorithms in IoT - A Study," in 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), Oct. 2020, pp. 332-336. doi: 10.1109/I-SMAC49090.2020.9243437.

[8] A. Srivastava, H. Pandey, A. Kumar, and M. Poonia, "Enhanced Hybrid Symmetric Cryptography for IoT Devices," in 2022 IEEE Conference on Interdisciplinary Approaches in Technology and Management for Social Innovation (IATMSI), Dec. 2022, pp. 1-4. doi: 10.1109/IATMSI56455.2022.10119338.

[9] "18_Paper_325-Blended_Cryptography_for_Secured_Data_Transfer.pdf". Accessed: February 3, 2024. [Online]. Available at: https://saiconference.com/Downloads/FTC2017/Proceedings/18_Paper_325-Blended_Cryptography_for_Secured_Data_Transfer.pdf

[10] T. B. Ogunseyi and O. M. Adedayo, "Cryptographic Techniques for Data Privacy in Digital Forensics", IEEE Access, vol. 11, pp. 142392-142410, 2023, doi: 10.1109/ACCESS.2023.3343360.

[11] K. Subramanian and F. L. John, "Dynamic Data Slicing in Multi Cloud Storage Using Cryptographic Technique," in 2017 World Congress on Computing and Communication Technologies (WCCCT), Feb. 2017, pp. 159-161. doi: 10.1109/WCCCT.2016.46.

[12] B. Al-Kasasbeh, "A Novel Secure Transposition Cipher Technique using Arbitrary Zigzag Patterns", IJACSA, vol. 13, n° 1, 2022, doi: 10.14569/IJACSA.2022.0130133.

[13] S. Chandra, S. Paira, S. S. Alam, and G. Sanyal, "A comparative survey of Symmetric and Asymmetric Key Cryptography," in 2014 International Conference on Electronics, Communication and Computational Engineering (ICECCE), Hosur, Tamilnadu, India: IEEE, Nov. 2014, pp. 83-93. doi: 10.1109/ICECCE.2014.7086640.

[14] A. Abdullah, A. Ismael, A. Rashid, A. Abou-Elnour, and M. Tarique, "Real Time Wireless Health Monitoring Application Using Mobile Devices," IJCNC, vol. 7, n° 3, pp. 13-30, May 2015, doi: 10.5121/ijcnc.2015.7302.

[15] M. A. A. Razali, M. Kassim, N. A. Sulaiman, and S. Saaidin, "A ThingSpeak IoT on Real Time Room Condition Monitoring System," in 2020 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS), June 2020, pp. 206-211. doi: 10.1109/I2CACIS49202.2020.9140127.

[16] V. D. Soni, "An IoT Based Patient Health Monitoring System".

[17] I. K. Dutta, B. Ghosh, and M. Bayoumi, "Lightweight Cryptography for Internet of Insecure Things: A Survey," in 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA: IEEE, Jan. 2019, pp. 0475-0481. doi: 10.1109/CCWC.2019.8666557.

[18] R. Sharma and S. Pansare, "ANALYSIS OF SYMMETRIC KEY CRYPTOGRAPHIC ALGORITHMS", vol. 04, n° 02.

[19] S. Surendran, A. Nassef, and B. D. Beheshti, "A survey of cryptographic algorithms for IoT devices," in 2018 IEEE Long Island Systems, Applications and Technology Conference (LISAT), May 2018, pp. 1-8. doi: 10.1109/LISAT.2018.8378034.

[20] M. A. Bahnasawi et al, "ASIC-oriented comparative review of hardware security algorithms for internet of things applications", in 2016 28th International Conference on Microelectronics (ICM), Dec. 2016, pp. 285-288. doi: 10.1109/ICM.2016.7847871.

[21] S. Agwa, E. Yahya, and Y. Ismail, "Power efficient AES core for IoT constrained devices implemented in 130nm CMOS," in 2017 IEEE International Symposium on Circuits and Systems (ISCAS), May 2017, pp. 1-4. doi: 10.1109/ISCAS.2017.8050361.

[22] D.-H. Bui, D. Puschini, S. Bacles-Min, E. Beigné, and X.-T. Tran, "Ultra low-power and low-energy 32-bit datapath AES architecture for IoT applications," in 2016 International Conference on IC Design and Technology (ICICDT), June 2016, pp. 1-4. doi: 10.1109/ICICDT.2016.7542076.

[23] M. Lu, A. Fan, J. Xu, and W. Shan, "A Compact, Lightweight and Low-Cost 8-Bit Datapath AES Circuit for IoT Applications in 28nm CMOS," in 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/ 12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), August 2018, pp. 1464-1469. doi: 10.1109/TrustCom/BigDataSE.2018.00204.

[24] A. Maier, A. Sharp, and Y. Vagapov, "Comparative analysis and practical implementation of the ESP32 microcontroller module for the internet of things," in 2017 Internet Technologies and Applications (ITA), Sept. 2017, pp. 143-148. doi: 10.1109/ITECHA.2017.8101926.

[25] P. Macheso, S. Chisale, C. Daka, N. Dzupire, J. Mlatho, and D. Mukanyirigira, "Design of Standalone Asynchronous ESP32 Web-Server for Temperature and Humidity Monitoring", in 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), March 2021, pp. 635-638. doi: 10.1109/ICACCS51430.2021.9441845.

[26] D. P. Wiliyanto et al, "Design and Development of a Mobile Goods Transfer Robot Using Waypoint GPS Navigation Method," in 2023 International Conference on Sustainable Emerging Innovations in Engineering and Technology (ICSEIET), Sept. 2023, pp. 517-522. doi: 10.1109/ICSEIET58677.2023.10303290.

[27] Y. X. Wu, D. Liu, and X. H. Kuang, "A temperature detecting system based on DS18B20," Advanced Materials Research, vol. 328, p. 1806-1809, 2011.

[28] Z. J. Liu, "Multi point temperature measurement system based on DS18B20", Advanced Materials Research, vol. 756, pp. 556-559, 2013.

[29] H. Shen, J. Fu, and Z. Chen, "Embedded system of temperature testing based on DS18B20", in 2006 International Technology and Innovation Conference (ITIC 2006), IET, 2006, pp. 2223-2226. Accessed: February 10, 2024. [Online]. Available at: https://ieeexplore.ieee.org/abstract/document/4752381/

[30] H. Xu, W. Wang, W. Deng, and Z. Lun, "Design of portable refrigerator based on DS18B20 temperature sensor," in 2023 2nd International Symposium on Sensor Technology and Control (ISSTC), August 2023, pp. 32-36. doi: 10.1109/ISSTC59603.2023.10281139.

[31] M. Ishibashi et al, "Increasing the Sensitivity of Piezoelectric Pulse Wave Sensors for Pulse Wave Propagation Measurement", IEEE Sensors Letters, vol. 7, nº 9, p. 1-4, Sept. 2023, doi: 10.1109/LSENS.2023.3308120.

[32] J. Chen et al, "Finger-Worn Dense Pressure-Sensor Array for Arterial Pulse Acquisition", in 2021 21st International Conference on Solid-State Sensors, Actuators and Microsystems (Transducers), June 2021, pp. 1468-1471. doi: 10.1109/Transducers50396.2021.9495720.

[33] S. R. Dasarapalli, R. K. Panneerselvam, S. Annavarapu, and K. R. Gopireddy, "Three-Lead ECG and Heartrate Variability Using AD8232," in 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), July 2023, pp. 1-6. doi: 10.1109/ICCCNT56998.2023.10306769.

[34] A. S. Prasad and N. Kavanashree, "ECG Monitoring System Using AD8232 Sensor," in 2019 International Conference on Communication and Electronics Systems (ICCES), July 2019, pp. 976-980. doi: 10.1109/ICCES45898.2019.9002540.

[35] M. W. Gifari, H. Zakaria, and R. Mengko, "Design of ECG Homecare:12-lead ECG acquisition using single channel ECG device developed on AD8232 analog front end," in 2015 International Conference on Electrical Engineering and Informatics (ICEEI), August 2015, pp. 371-376. doi: 10.1109/ICEEI.2015.7352529.

[36] H. Güvenç, "Wireless ECG Device with Arduino", in 2020 Medical Technologies Congress (TIPTEKNO), Nov. 2020, p. 1-4. doi: 10.1109/TIPTEKNO50054.2020.9299248.

[37] H. Zhao, C.-H. Su, J.-N. Xu, and J.-W. Zhu, "An OLED Efficiency Enhancement Strategy to Accelerate the Application of Flexible Optoelectronic Devices in the Biomedical Field," in 2022 16th ICME International Conference on Complex Medical Engineering (CME), Nov. 2022, pp. 95-98. doi: 10.1109/CME55444.2022.10063272.

[38] S. Kunić and Z. Šego, "OLED technology and displays", in Proceedings ELMAR-2012, Sept. 2012, pp. 31-35. Accessed: February 11, 2024. [Online]. Available at: https://ieeexplore.ieee.org/document/6338465/

[39] "Generating an AES key - Mbed TLS documentation. Accessed: March 5, 2024. [Online]. Available at: https://mbed-tls.readthedocs.io/en/latest/kb/how-to/generate-an-aes-key/

[40] F. J. Pal, "josephpal/esp32-Encrypt". March 5, 2024. Accessed: March 5, 2024. [Online]. Available at: https://github.com/josephpal/esp32-Encrypt