

الجمهورية الجزائرية الديمقراطية الشعبية

**REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE**

وزارة التعليم العالي والبحث العلمي

**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**

جامعة أبي بكر بلقايد - تلمسان

Université Aboubakr Belkaïd – Tlemcen –

Faculté de TECHNOLOGIE



## **MEMOIRE**

Présenté pour l'obtention du **diplôme** de **MASTER**

**En** : Télécommunications

**Spécialité** : Réseaux et Télécommunications

**Par** : Abdellaoui Seyf Eddine & Abdelali Djelloul

### **Sujet**

***Conception et réalisation d'un système IoT pour le suivi des patients cardiaques***

Soutenu publiquement, le 29/06/2022, devant le jury composé de :

Mr. R. MERZOUGUI

PR

Univ. Tlemcen

Président

Mr. R. BORSALI

PR

Univ. Tlemcen

Examineur

Mr. M. HADJILA

MCA

Univ. Tlemcen

Encadrant

Année universitaire : 2021 / 2022

# *Remerciements*



Nous adressons nos remerciements premièrement à ALLAH le tout puissant pour la volonté, la force, le courage, et la patience qu'il nous a donné durant toute la période d'études.

Nous exprimons nos sincères remerciements et nos profondes gratitudee à notre encadrant Mr Hadjila Mourad, pour sa patience, sa disponibilité, et ses conseils et orientations au long de la réalisation de ce travail.

Nos remerciements les plus respectueux à tous les membres du jury Mr Merzougui Rachid et Mr Borsali Riad, pour avoir accepté d'examiner et de juger ce travail.

Enfin, nous tenons à remercier nos enseignants et collègues à l'Université de Tlemcen, nos familles, nos proches, nos amis et tous ceux qui ont contribué de près ou de loin à notre succès.

# *Dédicaces*



Je dédie ce travail à :

Mes parents,

Ma famille,

Mes ami(e)s,

Mon Université,

Tous ceux qui m'aiment et que j'aime.

*Djelloul Abdelali*

# *Dédicaces*



Je dédie ce modeste travail :

Aux personnes les plus chères de ma vie, ma mère "Que Dieu te protège", et mon père "Que Dieu ait pitié de lui", pour exprimer ma gratitude éternelle pour tout ce qu'ils m'ont donné dans ma vie.

A mes chères sœurs, qui je souhaite un avenir plein de bonté et de réussite.

A mon binôme.

A ma famille, mes proches.

A mes amis et collègues.

*Seyf Eddine Abdellaoui*

## Résumé

Aujourd'hui l'IoT couvrira un large éventail d'applications et touchera quasiment à tous les domaines que nous affrontons au quotidien tels que les villes intelligentes, la domotique, l'agriculture intelligente, la santé, le commerce, etc. Dans le domaine de la santé, l'IoT utilise des dispositifs électroniques et des capteurs médicaux, en particulier pour la surveillance à distance des patients. Les pathologies cardiaques constituent la cause principale de mortalité la plus courante dans le monde. Cette maladie nécessite une attention considérable de la part du corps médical. L'objectif de ce travail est de concevoir et réaliser un système de télésurveillance médicale basé sur l'Internet des objets (IoT) afin d'assurer une surveillance continue des personnes cardiaques tout en gardant un contact permanent entre le patient et son médecin. Ce système est composé d'un module WEMOS D1 Mini (ESP8266), d'un capteur ECG (AD8232) et d'un module GSM/GPRS/GPS (SIM808).

**Mots clés :** Cardiopathie, Internet of Things, Arduino, ESP8266, SIM808, Capteur cardiaque AD8232.

## ملخص

تغطي إنترنت الأشياء اليوم مجموعة واسعة من التطبيقات وتؤثر تقريبًا على جميع المجالات التي نواجهها يوميًا مثل المدن الذكية، وأتمتة المنازل، والزراعة الذكية، والصحة، والتجارة، وما إلى ذلك. في مجال الرعاية الصحية، تستخدم إنترنت الأشياء الأجهزة الإلكترونية وأجهزة الاستشعار الطبية، خاصةً لمراقبة المريض عن بُعد. أمراض القلب هي السبب الأكثر شيوعًا للوفاة في جميع أنحاء العالم. يتطلب هذا المرض اهتمامًا كبيرًا من قبل مهنة الطب. الهدف من هذا العمل هو تصميم وتنفيذ نظام مراقبة طبية عن بعد يعتمد على إنترنت الأشياء (IoT) من أجل ضمان المراقبة المستمرة لمرضى القلب مع الحفاظ على اتصال دائم بين المريض وطبيبه. يتكون هذا النظام من وحدة WEMOS D1 Mini (ESP8266)، ومستشعر ECG (AD8232) ووحدة GSM / GPRS / GPS (SIM808).

الكلمات المفتاحية : أمراض القلب، إنترنت الأشياء، أردوينو، ESP8266، SIM808، مستشعر القلب AD8232.

## **Abstract**

Today the IoT will cover a wide range of applications and will affect almost all the areas that we face on a daily basis such as smart cities, home automation, smart agriculture, health, commerce, etc. In healthcare, the IoT uses electronic devices and medical sensors, especially for remote patient monitoring. Heart disease is the most common cause of death worldwide. This disease requires considerable attention from the medical profession. The objective of this work is to design and implement a medical telemonitoring system based on the Internet of Things (IoT) in order to ensure continuous monitoring of cardiac patients while maintaining permanent contact between the patient and his doctor. This system is composed of a WEMOS D1 Mini module (ESP8266), an ECG sensor (AD8232) and a GSM/GPRS/GPS module (SIM808).

**Keywords :** Cardiology, Internet of Things, Arduino, ESP8266, SIM808, AD8232 heart sensor.

# Table des matières

<i>Remerciements</i> .....	I
<i>Dédicaces</i> .....	II
<i>Résumé</i> .....	IV
<i>Table des matières</i> .....	VI
<i>Liste des Figures</i> .....	X
<i>Liste des Tableaux</i> .....	XII
<i>Liste des Symboles et Abréviations</i> .....	XIII
<i>Introduction Générale</i> .....	1

## *Chapitre I :*

### *Notions sur l'Internet des objets*

<b>I.1. Introduction</b> .....	2
<b>I.2. Evolution historique de l'Internet des objets</b> .....	2
<b>I.3. Définition de l'Internet des Objets</b> .....	3
<b>I.4. Les composants d'un modèle d'Internet des objets</b> .....	3
<b>I.4.1. La couche 1</b> .....	4
<b>I.4.2. La couche 2</b> .....	4
<b>I.4.3. La couche 3</b> .....	4
<b>I.5. Fonctionnement de l'Internet des Objets</b> .....	5
<b>I.5.1. Les technologies de courte portée</b> .....	5
<b>I.5.1.1. Le protocole NFC</b> .....	5
<b>I.5.1.2. Bluetooth</b> .....	6
<b>I.5.1.3. Zigbee</b> .....	6
<b>I.5.2. Les technologies de moyenne portée</b> .....	7
<b>I.5.2.1. WiFi</b> .....	7
<b>I.5.2.2. Bluetooth Low Energy</b> .....	8
<b>I.5.3. Les technologies de longue portée</b> .....	9
<b>I.5.3.1. Réseaux cellulaires mobiles</b> .....	9
<b>I.5.3.2. Réseaux radio bas-débit</b> .....	10
<b>I.5.3.3. Réseaux propriétaires</b> .....	12
<b>I.6. Domaines d'application de l'Internet des Objets</b> .....	12
<b>I.6.1. Ville intelligente</b> .....	13
<b>I.6.2. La domotique</b> .....	15

## *Table des matières*

---

<b>I.6.3.</b>	<b>Agriculture intelligente .....</b>	<b>15</b>
<b>I.6.4.</b>	<b>Industrie intelligente .....</b>	<b>16</b>
<b>I.6.5.</b>	<b>Santé intelligente .....</b>	<b>17</b>
<b>I.6.5.1.</b>	<b>Les objets connectés pour se challenger .....</b>	<b>17</b>
<b>I.6.5.2.</b>	<b>Les objets connectés pour se surveiller .....</b>	<b>17</b>
<b>I.6.5.3.</b>	<b>Les applications / objets connectés dans l'exercice médical .....</b>	<b>18</b>
<b>I.7.</b>	<b>Les objets intelligents dans l'IoT .....</b>	<b>18</b>
<b>I.8.</b>	<b>L'importance des objets connectés dans la santé .....</b>	<b>19</b>
<b>I.9.</b>	<b>Les enjeux de l'Internet des Objets .....</b>	<b>20</b>
<b>I.9.1.</b>	<b>Risques d'utilisation des objets connectés .....</b>	<b>20</b>
<b>I.9.2.</b>	<b>Sécurisation de l'appareil .....</b>	<b>20</b>
<b>I.9.2.1.</b>	<b>PKI et certificats numériques CA .....</b>	<b>21</b>
<b>I.9.2.2.</b>	<b>Sécurité des réseaux .....</b>	<b>21</b>
<b>I.9.2.3.</b>	<b>Sécurité des API .....</b>	<b>21</b>
<b>I.9.3.</b>	<b>Types d'environnement cloud .....</b>	<b>22</b>
<b>I.9.4.</b>	<b>Sécurisation du cloud .....</b>	<b>22</b>
<b>I.9.4.1.</b>	<b>Solutions CrowdStrike de sécurité du cloud .....</b>	<b>22</b>
<b>I.9.5.</b>	<b>Gestion du cycle de vie d'un objet connecté .....</b>	<b>22</b>
<b>I.10.</b>	<b>Le Web des objets .....</b>	<b>24</b>
<b>I.11.</b>	<b>Conclusion .....</b>	<b>24</b>

### *Chapitre II :*

#### *Présentation du système IoT réalisé*

<b>II.1.</b>	<b>Introduction .....</b>	<b>25</b>
<b>II.2.</b>	<b>Présentation du système .....</b>	<b>25</b>
<b>II.3.</b>	<b>Matériel requis .....</b>	<b>26</b>
<b>II.3.1.</b>	<b>La carte Arduino .....</b>	<b>26</b>
<b>II.3.1.1.</b>	<b>Définition .....</b>	<b>26</b>
<b>II.3.1.2.</b>	<b>Les gammes de la carte Arduino .....</b>	<b>27</b>
<b>II.3.2.</b>	<b>Le WEMOS D1 Mini (ESP8266) .....</b>	<b>28</b>
<b>II.3.2.1.</b>	<b>Définition .....</b>	<b>28</b>
<b>II.3.2.2.</b>	<b>Caractéristiques .....</b>	<b>29</b>
<b>II.3.2.3.</b>	<b>Les broches de WEMOS D1 Mini .....</b>	<b>29</b>
<b>II.3.3.</b>	<b>Le module GSM/GPRS/GPS (SIM808) .....</b>	<b>31</b>
<b>II.3.3.1.</b>	<b>Définition .....</b>	<b>31</b>
<b>II.3.3.2.</b>	<b>Interfaces .....</b>	<b>31</b>
<b>II.3.3.3.</b>	<b>Caractéristiques générales .....</b>	<b>32</b>
<b>II.3.3.4.</b>	<b>Caractéristiques GPS .....</b>	<b>33</b>

---



## *Table des matières*

---

II.3.3.5.	Le montage de la carte WEMOS et le module SIM808.....	33
II.3.3.6.	Les commandes AT.....	34
II.3.4.	Capteur de fréquence cardiaque ECG.....	34
II.3.4.1.	Description du signal électrocardiogramme (ECG).....	34
II.3.4.2.	Le module AD8232 (capteur ECG).....	37
II.3.4.3.	Caractéristiques de capteur ECG.....	38
II.3.4.4.	Connexion du capteur ECG et la carte WEMOS D1 Mini.....	39
II.3.5.	LEDs et résistances protectrices.....	39
II.3.6.	Plaque d'essai et câbles de démarrage (Jumper cables).....	40
II.4.	Logiciels et plateformes requis.....	41
II.4.1.	L'environnement de développement Arduino.....	41
II.4.1.1.	Description du logiciel Arduino IDE.....	41
II.4.1.2.	Syntaxe du langage Arduino.....	43
II.4.2.	Application web.....	45
II.4.2.1.	HTML.....	46
II.4.2.2.	CSS.....	46
II.4.2.3.	JavaScript.....	46
II.4.3.	Cloud Computing.....	46
II.4.3.1.	Définition.....	46
II.4.3.2.	Caractéristiques.....	47
II.4.3.3.	Firebase.....	48
II.5.	Conclusion.....	49

### *Chapitre III :*

#### *La réalisation pratique du système IoT*

III.1.	Introduction.....	50
III.2.	Montage du système.....	50
III.3.	Installation de la carte ESP8266 dans l'Arduino IDE.....	52
III.4.	L'algorithme global du fonctionnement du système.....	54
III.4.1.	Importation de bibliothèques.....	54
III.4.2.	La fonction setup.....	54
III.4.3.	La fonction loop.....	54
III.4.4.	L'organigramme global.....	55
III.5.	L'algorithme d'acquisition du signal ECG.....	56
III.6.	L'algorithme d'envoyer les valeurs à Firebase Cloud.....	56
III.7.	L'algorithme d'acquisition du GPS et d'envoyer les SMS.....	57
III.8.	Création d'un projet Firebase.....	57
III.9.	Création d'application web.....	59

---

## *Table des matières*

---

---

<b>III.10. Résultats obtenus</b> .....	60
<b>III.11. Conclusion</b> .....	66
<i>Conclusion Générale</i> .....	67
<i>Bibliographie</i> .....	68
<i>Annexe</i> .....	70

## Liste des Figures

Figure I.1 : Nombre d'appareils connectés dans le monde de 2015 à 2025 [3] .....	3
Figure I.2 : Les composants de l'IoT .....	4
Figure I.3 : Badge équipé d'une puce RFID. ....	5
Figure I.4 : Applications Bluetooth .....	6
Figure I.5 : Applications Zigbee.....	7
Figure I.6 : Architecteur de IEEE 802.11 .....	8
Figure I.7 : Application mobile BLE .....	9
Figure I.8 : Localiser un camion avec GPS à l'aide de GSM réseau cellulaire .....	10
Figure I.9 : Architecture d'un réseau SIGFOX .....	11
Figure I.10 : Une architecture de réseau LoRaWAN typique.....	12
Figure I.11 : Domaines d'application de l'Internet des objets.....	13
Figure I.12 : Smart City .....	14
Figure I.13 : Maison intelligente .....	15
Figure I.14 : Exemple de Agriculture intelligente.....	16
Figure I.15 : L'industrie 4.0 .....	17
Figure I.16 : Santé intelligente .....	18
Figure I.17 : Cycle de vie d'un objet connecté.....	23
Figure I.18 : L'architecture du Web of Things (WoT).....	24
Figure II.1 : Conception globale du système proposé.....	26
Figure II.2 : Exemples du carte Arduino .....	27
Figure II.3 : Les gammes de la carte Arduino .....	27
Figure II.4 : Le module WEMOS D1 Mini .....	29
Figure II.5 : Les broches de module WEMOS D1 Mini .....	30
Figure II.6 : Le module GSM/GPRS/GPS (SIM808) .....	31
Figure II.7 : Interfaces du SIM808 .....	32
Figure II.8 : Communication du carte WEMOS et carte SIM808 .....	33
Figure II.9 : Le signal d'électrocardiogramme (ECG) .....	35
Figure II.10 : Capteur de fréquence cardiaque ECG (module AD8232) .....	37
Figure II.11 : L'emplacement des électrodes dans le corps.....	38
Figure II.12 : Connexion du capteur ECG et la carte WEMOS D1 Mini.....	39
Figure II.13 : Les LEDs et Les Résistances protectrices .....	40
Figure II.14 : Plaquette d'essai sans soudure et câbles Jumper .....	40
Figure II.15 : La fenêtre principale du logiciel Arduino IDE.....	41
Figure II.16 : Interface Arduino IDE.....	42
Figure II.17 : Fenêtre de moniteur série .....	43
Figure II.18 : Fenêtre de Traceur série .....	43
Figure II.19 : Cloud Computing .....	47
Figure II.20 : Firebase cloud .....	48

## *Liste des Figures*

---

---

<b>Figure III.1 : Circuit du Système</b> .....	50
<b>Figure III.2 : Le montage de notre système</b> .....	51
<b>Figure III.3 : Fenêtre de Préférences de l'Arduino IDE</b> .....	52
<b>Figure III.4 : Fenêtre de gestionnaire de cartes de l'Arduino IDE</b> .....	53
<b>Figure III.5 : Sélection de la carte WEMOS D1 mini.</b> .....	53
<b>Figure III.6 : Organigramme global du système</b> .....	55
<b>Figure III.7 : Organigramme de la fonction de calcul de battement</b> .....	56
<b>Figure III.8 : Organigramme de la fonction qui envoie les valeurs à Firebase Cloud</b> .....	57
<b>Figure III.9 : Organigramme de la fonction d'acquisition du GPS et d'envoi des SMS</b> .	57
<b>Figure III.10 : Création d'un projet Firebase</b> .....	58
<b>Figure III.11 : Créer un projet (Étape 1)</b> .....	58
<b>Figure III.12 : Créer un projet (Étape 2)</b> .....	58
<b>Figure III.13 : Objet de configuration Firebase</b> .....	59
<b>Figure III.14 : Modifier les règles pour Realtime Database</b> .....	60
<b>Figure III.15 : Affichage des paramètres de point d'accès sur le moniteur série</b> .....	60
<b>Figure III.16 : Connexion au point d'accès fourni par le WEMOS</b> .....	61
<b>Figure III.17 : Interface de configuration du WEMOS</b> .....	62
<b>Figure III.18 : Résultats de la fonction "setup" affichés sur le moniteur série</b> .....	62
<b>Figure III.19 : Lecteur du signal ECG sur le traceur série</b> .....	63
<b>Figure III.20 : Transfert des données en temps réel vers Firebase</b> .....	63
<b>Figure III.21 : Interface utilisateur de l'application web version mobile</b> .....	64
<b>Figure III.22 : Interfaces utilisateur de l'application web version PC</b> .....	65
<b>Figure III.23 : Alerte par SMS</b> .....	66

## *Liste des Tableaux*

<b>Tableau I.1 : Les normes WiFi et leurs caractéristiques.....</b>	<b>8</b>
<b>Tableau II.1 : Chronologie des sorties de cartes Arduino .....</b>	<b>28</b>
<b>Tableau II.2 : Les caractéristiques de la carte WEMOS D1 Mini.....</b>	<b>29</b>
<b>Tableau II.3 : Les broches de module WEMOS D1 Mini.....</b>	<b>30</b>
<b>Tableau II.4 : Commande AT et leur description .....</b>	<b>34</b>
<b>Tableau II.5 : Les valeurs habituelles des différentes composantes du signal ECG.....</b>	<b>36</b>
<b>Tableau II.6 : Connexion du capteur ECG et la carte WEMOS D1 Mini .....</b>	<b>39</b>
<b>Tableau III.1 : Connection de différentes broches de la carte WEMOS .....</b>	<b>51</b>

## *Liste des Symboles et Abréviations*

<b>AP</b>	Access Point.
<b>API</b>	Application Programming Interface.
<b>BIM</b>	Building Information Modeling.
<b>BLE</b>	Bluetooth Low Energy.
<b>Bluetooth SIG</b>	Bluetooth Special Interest Group.
<b>BPM</b>	Beats per Minute
<b>BSS</b>	Basic Service Set.
<b>BSSID</b>	Basic Service Set Identifier.
<b>CGNS</b>	CFD General Notation System.
<b>CNS</b>	Conférence Nationale de Santé.
<b>Commande AT</b>	Commande ATtention.
<b>CSPM</b>	Cloud Security Posture Management.
<b>CVC</b>	chauffage, ventilation et climatisation.
<b>DC</b>	Direct Current.
<b>DDoS</b>	Distributed Denial of Service.
<b>DevOps</b>	Development et Operations.
<b>E/S</b>	Entrées/Sorties.
<b>EAP</b>	Extensible Authentication Protocol.
<b>ECG</b>	ÉlectroCardioGramme.
<b>EEPROM</b>	Electrically-Erasable Programmable Read-Only Memory.
<b>EHpad</b>	Établissement d'hébergement pour personnes âgées dépendantes.
<b>GMAO</b>	Gestion de Maintenance Assistée par Ordinateur.
<b>GND</b>	Ground.
<b>GNSS</b>	Global Navigation Satellite System.
<b>GPIO</b>	General Purpose Input/Output.
<b>GPRS</b>	General Packet Radio Service.
<b>GPS</b>	Global Positioning System.
<b>GRU</b>	Gestion de la Relation Usager.
<b>GSM</b>	Global System for Mobile Communications.
<b>GTI</b>	Gestion Techniques des Interventions.
<b>GUI</b>	Graphical User Interface.
<b>HIP-DEX</b>	Host Identity Protocol Diet Exchange.

## *Liste des Symboles et Abréviations*

---

<b><i>HS</i></b>	Handle System.
<b><i>HTTP</i></b>	HyperText Transfer Protocol.
<b><i>I2C</i></b>	Inter-Integrated Circuit.
<b><i>I2S</i></b>	Integrated Interchip Sound.
<b><i>IaaS</i></b>	Infrastructure as a Service.
<b><i>ICT</i></b>	Information and Communication Technologies.
<b><i>IDE</i></b>	Integrated Development Environment.
<b><i>IMC</i></b>	l'indice de masse corporelle.
<b><i>IoT</i></b>	Internet of Things.
<b><i>IP</i></b>	Internet Protocol.
<b><i>ISM</i></b>	Bande industrielle, scientifique et médicale.
<b><i>LDAP</i></b>	Lightweight Directory Access Protocol.
<b><i>LED</i></b>	Light-Emitting Diode.
<b><i>LoRa</i></b>	Long Range.
<b><i>LoRaWAN</i></b>	Long Range Wide Area Network.
<b><i>LTE</i></b>	Long Term Evolution.
<b><i>M2M</i></b>	communication Machine to Machine.
<b><i>MAC</i></b>	Media Access Control.
<b><i>MISO</i></b>	Master In Slave Out.
<b><i>MOSI</i></b>	Master Out Slave In.
<b><i>NFC</i></b>	Near Field Communication.
<b><i>NMEA</i></b>	National Marine Electronics Association.
<b><i>OS</i></b>	Operating System.
<b><i>PaaS</i></b>	Platform as a Service.
<b><i>PANA</i></b>	Protocol for Carrying Authentication for Network Access.
<b><i>PC</i></b>	Personal Computer.
<b><i>PCM</i></b>	Pulse Code Modulation.
<b><i>PWM</i></b>	Pulse Width Modulation.
<b><i>RFID</i></b>	Radio Frequency Identification.
<b><i>RFID-HF</i></b>	Radio Frequency Identification High Frequency
<b><i>RX</i></b>	Receiver.
<b><i>SaaS</i></b>	Software as a service.
<b><i>SCK</i></b>	Source Clock.
<b><i>SDA</i></b>	Serial Data Acces.
<b><i>SIM</i></b>	Subscriber Identity Module.
<b><i>SMS</i></b>	short message system.
<b><i>SoC</i></b>	System on a Chip.

## *Liste des Symboles et Abréviations*

---

<b><i>SOL</i></b>	Security Of Life.
<b><i>SPI</i></b>	Serial Peripheral Interface.
<b><i>SS</i></b>	Slave Select.
<b><i>TX</i></b>	Transmitter.
<b><i>UART</i></b>	Universal Asynchronous Receiver Transmitter.
<b><i>UNB</i></b>	Ultra narrow band.
<b><i>URL</i></b>	Uniform Resource Locator.
<b><i>USB</i></b>	Universal Serial Bus.
<b><i>W3C</i></b>	World Wide Web Consortium.
<b><i>WAN</i></b>	Wide Area Network.
<b><i>WDS</i></b>	Wireless Distribution System.
<b><i>WiFi</i></b>	Wireless Fidelity.
<b><i>WoT</i></b>	Web of Things.
<b><i>WPAN</i></b>	Wireless Personal Area Network.
<b><i>3G</i></b>	Third Generation.
<b><i>4G</i></b>	Fourth Generation.
<b><i>5G</i></b>	Five Generation.



# *Introduction Générale*

## *Introduction Générale*

---

L'Internet des objets est devenu un sujet intéressant de nos jours, il a atteint divers domaines de la vie humaine tels que l'agriculture, les soins de santé, la ville intelligente, l'énergie intelligente, etc.

Étant donné que de nombreuses personnes souffrent de maladies chroniques, la technologie de l'internet des objets dans le domaine de la santé, notamment dans ce qu'on appelle le domaine de la télésurveillance médicale, qui est une branche de la télémédecine (télécommunication + médecine), nous donne la possibilité de réduire le risque de ce type de maladie et assure un suivi continu du patient afin que le médecin soit toujours au courant de son état.

Dans le cadre de notre projet, nous adoptons la technologie Internet des objets pour réaliser un système de télésurveillance dédié aux patients cardiaques, basée sur l'acquisition du signal ECG d'un patient, compter le nombre de battements par minute, puis les envoyer en temps réel à la base de données distante.

Pour mettre en place le système, nous avons utilisé comme microprocesseur le module WEMOS D1 Mini qui permet d'effectuer les calculs nécessaires et de se connecter au réseau WiFi en même temps, le capteur de fréquence cardiaque et le module GSM/GPRS/GPS pour assurer la géolocalisation du patient et la transmission de SMS en cas d'anomalies observée chez le patient.

Notre travail est structuré comme suit :

Dans le premier chapitre, nous parlerons du concept de l'Internet des objets, de son fonctionnement et de son domaine d'application, en plus de la sécurité IoT.

Le deuxième chapitre présentera la description de notre système de télésurveillance tout en mettant en évidence les caractéristiques techniques et fonctionnelles de tous les matériels et logiciels utilisés dans le développement, avec une définition générale sur les signaux cardiaques (ECG).

Le troisième chapitre sera consacré à la conception et la réalisation pratique de notre projet IoT, ainsi qu'aux résultats obtenus.

# *Chapitre I :*

*Notions sur l'Internet des objets*

## **I.1. Introduction**

Dans ce chapitre, nous présenterons une idée générale sur l'IoT et tout ce qui est nécessaire pour mieux comprendre cette notion, en expliquant son architecture et les composants de base qui la constituent avec les différentes applications associées, tout en citant les enjeux les plus significatifs cités par ce concept sans oublier la sécurité et les risques d'utilisation de cette technologie et comment protéger les clients pour une meilleure utilisation. Ainsi, nous terminerons ce chapitre avec le Web des objets.

## **I.2. Evolution historique de l'Internet des objets**

L'Internet des objets est un concept qui incarne la vision de l'informatique omniprésente imaginée par Mark Weiser en 1991, dans laquelle la technologie s'estompe progressivement dans l'environnement de l'utilisateur, se fondant naturellement à l'intérieur des objets. La technologie n'est plus représentée par un ordinateur personnel objet unique, mais sous la forme d'appareils spécialisés et faciles à utiliser capables de communiquer sur de nombreux types de réseaux sans fil : liseuses, téléviseurs et montres connectées, ordinateurs de voiture, smartphones, etc.

À l'origine, le terme IoT a été inventé par Kevin Ashton en 1999 pour décrire des objets équipés de puces d'identification par radiofréquence (ou puces RFID). Chaque objet identifié de manière unique et générique peut alors être rattaché à un ensemble d'informations le concernant, ces dernières pouvant être lues par d'autres machines. Les fonctionnalités, l'état actuel et l'emplacement échangent alors autant de métadonnées que possible entre les objets, formant un nouveau réseau qui leur est dédié : l'Internet des objets.

Au fil du temps, cependant, le concept a évolué et s'est généralisé comme une approche qui consiste à connecter un grand nombre d'objets du quotidien au réseau Internet, à leur donner leur propre identité et à leur permettre de servir et de collecter des informations, indépendamment [1].

Au début 2008/2009 le nombre d'objets connectés a connu une augmentation qui n'était pas normale, le nombre de ces derniers dépassait la population humaine. Le chiffre de ces appareils a touché 27 milliards d'appareils connectés en 2020, qu'il s'agisse de capteurs, d'appareils électroménagers, de machines, d'éoliennes, de matériel médical ou de voitures. Les experts attendus cette technologie atteignaient 75 milliards « d'objets » d'ici 2025. Il y aura alors une dizaine d'objets connectés par personne sur Terre. Le marché de l'IoT devrait également exploser d'une valeur de 248 milliards de dollars en 2020 à près de 1 600 milliards en 2025 [2].

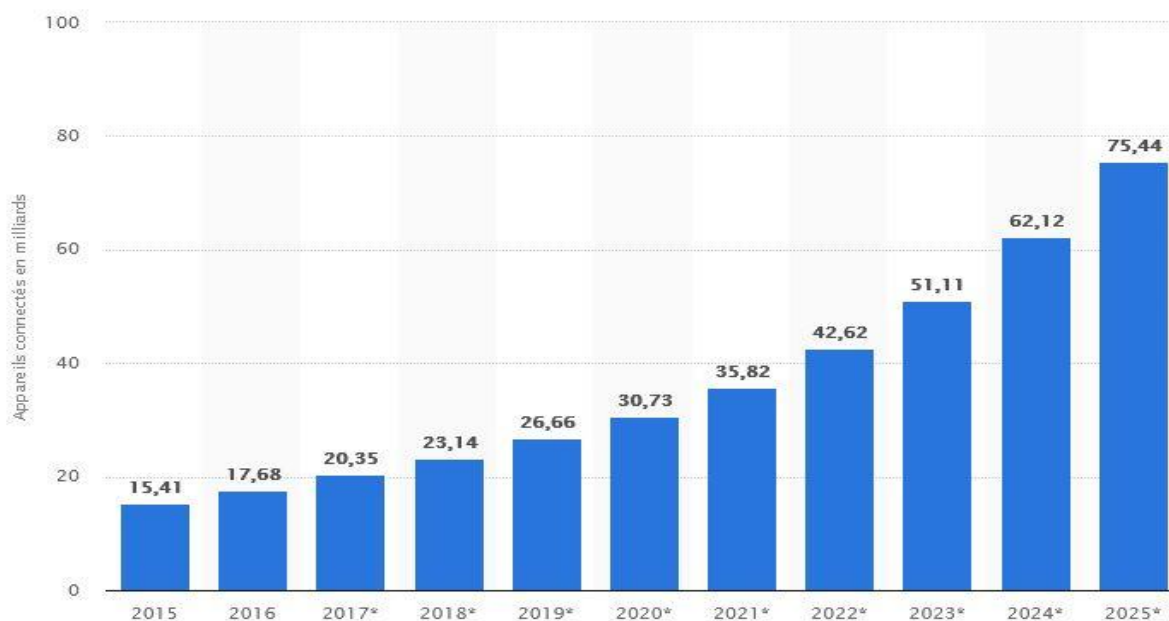


Figure I.1 : Nombre d'appareils connectés dans le monde de 2015 à 2025 [3]

### I.3. Définition de l'Internet des Objets

La technologie IoT englobe toute la propriété intellectuelle matérielle, les outils, les systèmes, les capteurs et les logiciels qui prennent en charge le développement d'appareils et d'applications IoT. Grâce à la technologie IoT, les objets du quotidien peuvent être rendus intelligents, des appareils médicaux aux smartphones, des montres aux caméras de sécurité, et des voitures aux lignes de production en usine. Les technologies IoT incluent également des outils de sécurité pour empêcher les attaques basées sur Internet sur les appareils en réseau et leurs applications [4].

Techniquement, l'IoT est une extension du système de nommage internet et traduit une convergence des identifiants numériques au sens où il est possible d'identifier de manière unifiée des éléments d'information numérique (adresses URL de sites web par exemple) et des éléments physiques. Mais l'identification est directe grâce à l'utilisation d'un système d'identification électronique (puces RFID, processeur et communication Bluetooth, etc.). Il n'y a pas besoin de saisir manuellement le code de l'objet. Le réseau s'étend jusqu'à lui et permet ainsi de créer une forme de passerelle entre les mondes physique et virtuel [5].

### I.4. Les composants d'un modèle d'Internet des objets

Un système IoT assemble de nombreux acteurs et composants technologiques. Il est composé d'objets connectés, de réseaux de communication sans fil, de plateformes de collecte, d'hébergement et de traitement des données.

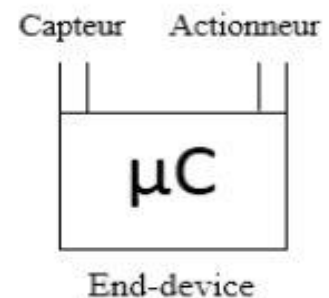
### I.4.1. La couche 1

La couche 1 est dédiée à collecter et agir en fonction des données. Elle se compose de :

- Les capteurs : Les capteurs sont des interfaces qui permettent de détecter, voire de mesurer certaines propriétés ou caractéristiques de l'environnement. Par exemple la température, lumière, humidité, proximité, présence, ondes cérébrales, etc.

- Les actionneurs : sont très largement utilisés dans les systèmes de contrôle, ils effectuent des actions à des moments précis.

- End-devices : Une petite carte électronique avec un microcontrôleur ( $\mu\text{C}$ ) intégré entre le capteur et l'actionneur, qui traite les données du capteur et envoie des commandes à l'actionneur.



### I.4.2. La couche 2

Cette couche contient des protocoles de communication et des passerelles pour le transfert d'informations entre End-devices et la base des données.

### I.4.3. La couche 3

Ce niveau comprend une plate-forme packagée basée sur le cloud pour traiter les données provenant des terminaux et les visualiser sous forme de tableaux ou de graphes pour les utilisateurs, plus une application software qui fournit des interfaces graphiques (GUI) pour la surveillance et le contrôle des End-devices.

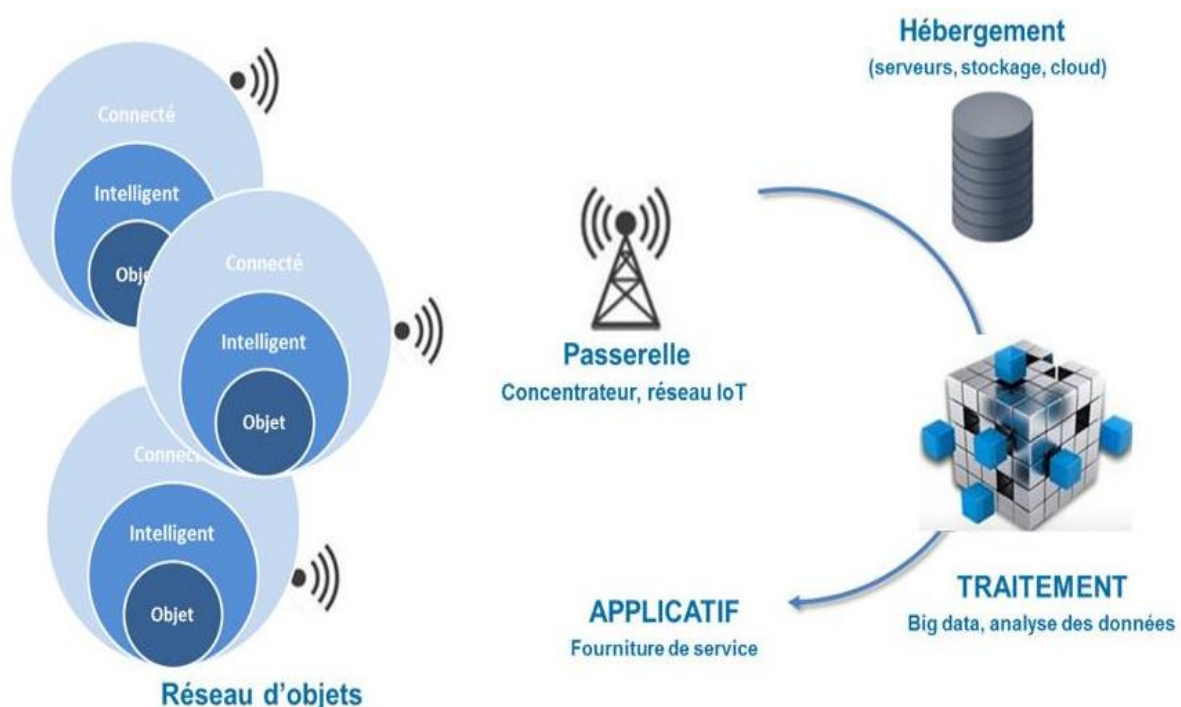


Figure I.2 : Les composants de l'IoT

## I.5. Fonctionnement de l'Internet des Objets

Les objets connectés nécessitent une communication entre eux pour connaître tous les détails de la zone et prendre des mesures et des actions justes et précises dans le temps. Il existe 3 moyens pour les connecter:

### I.5.1. Les technologies de courte portée

#### I.5.1.1. Le protocole NFC

Cette technologie, dite Near Filed, est destinée à la propagation de champs électromagnétiques à courte distance pour garder la sécurité et facilite la mise en œuvre et elle nécessite une consommation négligeable d'énergie, elle est basée sur la technologie RFID-HF (haute fréquence) à 13.56Mhz qui est principalement utilisée pour cartes à puce sans contact.

Le succès du Near Filed a conduit les chercheurs à mieux profiter de ce dernier en essayant de faire communiquer deux lecteurs ensemble à très courte distance de manière bidirectionnelle, nommés sur le nom de Peer to Peer, au fil du temps le champ proche a évolué et a attendu une opération consistant à émuler une fonction carte à puce sans contact ou une fonction lecteur RFID.

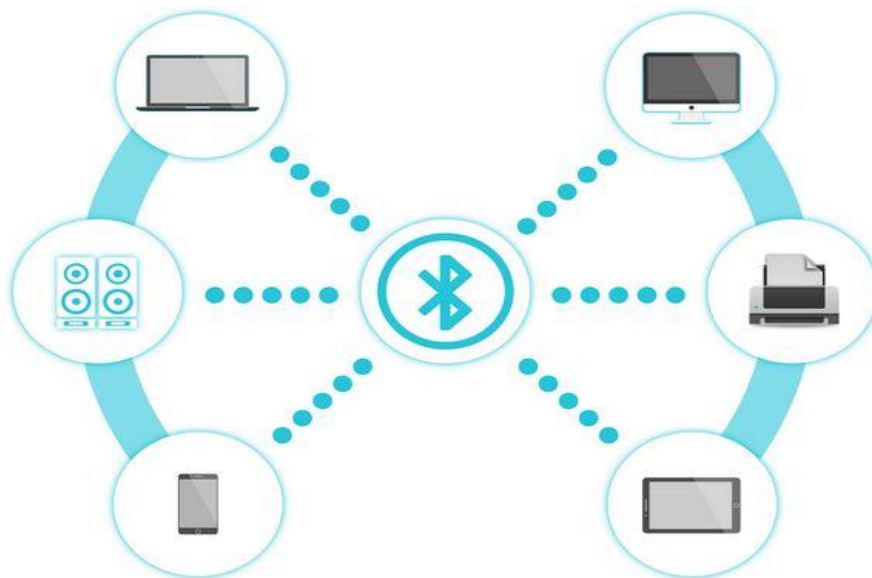
Ainsi est apparu le besoin de faire coexister tous ces différents besoins, tout en capitalisant sur le parc de lecteurs installés et sur les applications existantes, c'est la naissance du NFC [6].



*Figure I.3 : Badge équipé d'une puce RFID.*

### I.5.1.2. Bluetooth

La technologie sans fil Bluetooth est conçue pour fournir des solutions de connectivité à courte portée pour les appareils électroniques personnels, portables et dispositifs électroniques portatifs. Depuis mai 1998, le Bluetooth SIG a dirigé le développement de cette technologie en développant des spécifications industrielles ouvertes, y compris des protocoles et des cas d'utilisation, et des programmes de certification conçus pour garantir la valeur des produits Bluetooth aux utilisateurs finals. Il peut transmettre une petite quantité de données sur une courte distance, mais il est aussi très économe en énergie. A l'heure actuelle, dans la plupart des téléphones portables, le Bluetooth a pour fonction de réaliser la communication entre deux téléphones portables ou entre des téléphones portables et des objets connectés différents. Il a de nombreuses applications (casques de discussion téléphonique sans fil, montres intelligentes, moniteurs de fréquence cardiaque, haut-parleurs de diffusion de sons, etc. Le protocole est également utilisé sur des capteurs statiques appelés beamers pour mesurer le trafic, par exemple la transmission de la voix entre les équipements audio à faible puissance dans un magasin sur un rayon de l'ordre d'une dizaine de mètres.



*Figure I.4 : Applications Bluetooth*

### I.5.1.3. Zigbee

ZigBee est la norme de réseau maillé sans fil la plus populaire de l'industrie pour la connexion de capteurs, d'instrumentations et de systèmes de contrôle. ZigBee, une spécification pour la communication dans un réseau personnel sans fil (WPAN), a été appelée "l'Internet des objets". Théoriquement, votre cafetière compatible ZigBee peut communiquer avec votre grille-pain compatible ZigBee. ZigBee est un protocole ouvert, global, basé sur des paquets, conçu pour fournir



une architecture facile à utiliser pour des réseaux sans fil sécurisé, fiables et à faible consommation. ZigBee et IEEE 802.15.4 sont des normes de réseau sans fil à faible débit de données qui peuvent éliminer le câblage coûteux et sujet aux dommages dans les applications de contrôle industriel. Équipement de contrôle de flux ou de processus peut être placé n'importe où et toujours communiqué avec le reste du système. Il peut également être déplacé, car le réseau ne se soucie pas de l'emplacement physique d'un capteur, d'une pompe ou d'une vanne. La norme ZigBee améliore la norme IEEE 802.15.4 en fournissant une couche réseau simple et des profils d'application standard qui peuvent être utilisés pour créer des solutions électroniques grand public multifournisseurs interopérables.

Les avantages de cette technologie vont bien au-delà des applications ZigBee qui incluent la domotique, l'automatisation industrielle, la surveillance médicale, les capteurs basse consommation, le contrôle CVC, plus de nombreuses autres utilisations de contrôle et de surveillance [7].



Figure I.5 : Applications Zigbee

## I.5.2. Les technologies de moyenne portée

### I.5.2.1. WiFi

Le réseau WiFi est basé sur une architecture cellulaire, et chaque cellule, BSS (Basic Service Set), est contrôlée par une station de base radio (AP). Le BSS est identifié par le BSSID (Basic Service Set Identifier), qui est sur 6 octets et correspond à l'adresse MAC du point d'accès. Le réseau peut consister en une ou plusieurs unités autonomes, ou il peut s'agir d'une extension de l'Ethernet traditionnel. Dans ce dernier cas, différents points d'accès sont connectés à un

réseau de distribution qui fait office de réseau backbone (OS). Le lien entre les différents points d'accès peut être filaire ou sans fil (WDS), permettant des connexions haut débit sur des distances de 20 à 100 mètres. Il s'agit d'un réseau local sans fil très gourmand en énergie et ne convient qu'aux appareils qui sont connectés au secteur ou qui sont alimentés facilement et fréquemment. Il permet un transfert rapide de grandes quantités de données et peut-être utilisé avec IPv4 ou IPv6. Le Tableau I.1 décrit les différentes normes WiFi et leurs caractéristiques.

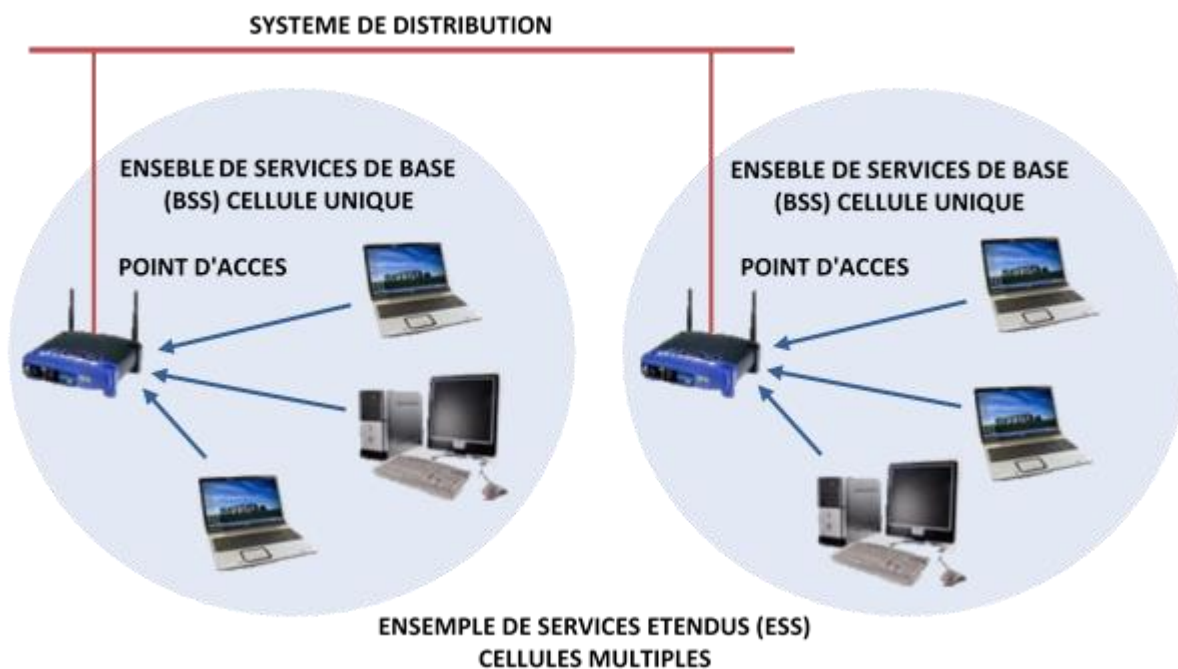


Figure I.6 : Architecture de IEEE 802.11

Protocole	Date de Normalisation	Fréquence GHz	Débit(Théorique) Mbit/s	Débit(Typique) Mbit/s	Portée intérieur m	Portée extérieur m
802.11a	1999	5	54	25	Environ 25	Environ 75
802.11b	1999	2.4	11	6.5	Environ 35	Environ 100
802.11g	2003	2.4	54	25	Environ 40	Environ 100
802.11n	2009	2.4 ou 5	300	100	Environ 80	Environ 150

Tableau I.1 : Les normes WiFi et leurs caractéristiques.

### I.5.2.2. Bluetooth Low Energy

Anciennement connu sous le nom de Wibree, puis enregistré sous le nom de Bluetooth Smart, c'est une technologie de transmission sans fil créée par Nokia en 2006, en tant que norme ouverte basée sur Bluetooth, c'est un complément à Bluetooth, mais pas un remplacement. Il a été intégré à la norme Bluetooth depuis la publication de la v4.0 de Bluetooth SIG en juin 2010. BLE ne peut transmettre

que des données limitées sur une distance moyenne de 60 mètres. La différence entre les appareils Bluetooth et BLE est la consommation d'énergie nécessaire à la communication, qui est dix fois inférieure à celle du Bluetooth. La figures I.7 montre l'application mobile Bluetooth Low Energy pour contrôler plusieurs dispositifs.



Figure I.7 : Application mobile BLE

### I.5.3. Les technologies de longue portée

#### I.5.3.1. Réseaux cellulaires mobiles

Les réseaux mobiles ont été construits sur une base d'architecture cellulaire qui permet d'utiliser plusieurs fois les mêmes fréquences sur une zone géographique, ils sont passés par plusieurs générations de communications téléphoniques, puis de transmission de messages vocaux et textuels jusqu'à l'attente des services data, internet mobile, très haut débit et objets connectés GSM, 3G, 4G et 5G [8].

Les réseaux cellulaires mobiles fournis par les opérateurs permettent le transfert de grandes quantités de données sur de longues distances et nécessitent l'installation d'une carte SIM dans les appareils connectés pour être identifiés sur le réseau. La succession de la norme de téléphonie mobile de première génération a permis d'augmenter progressivement les débits de communication jusqu'aux 20 gigabits de données par seconde pour la 5G. Des débits jusqu'à 3 à 4 fois plus rapides que la 4G. La figure I.8 permet de localiser un camion avec GPS à l'aide de GSM réseau cellulaire.

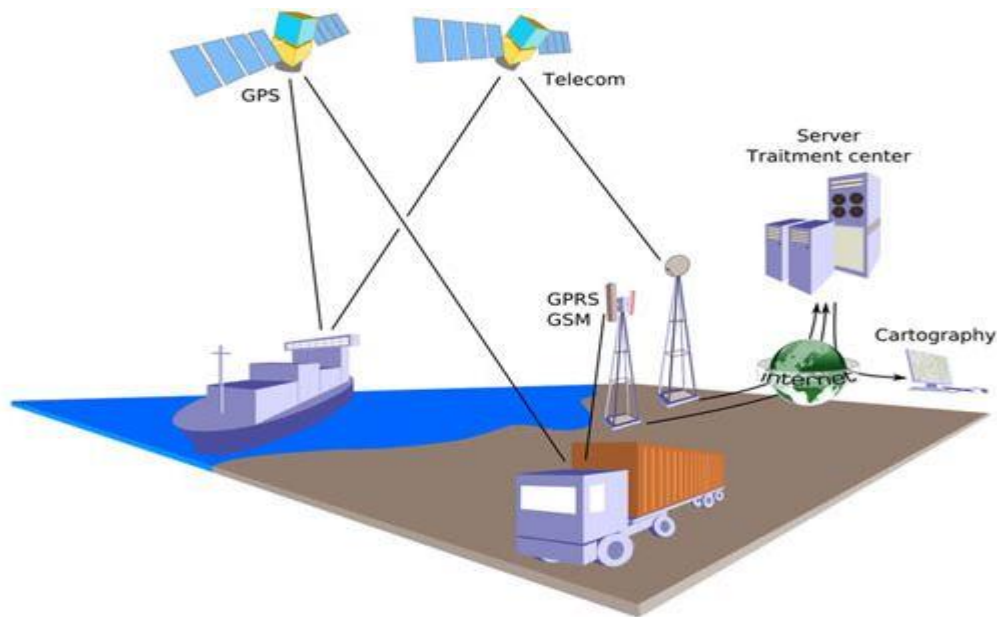
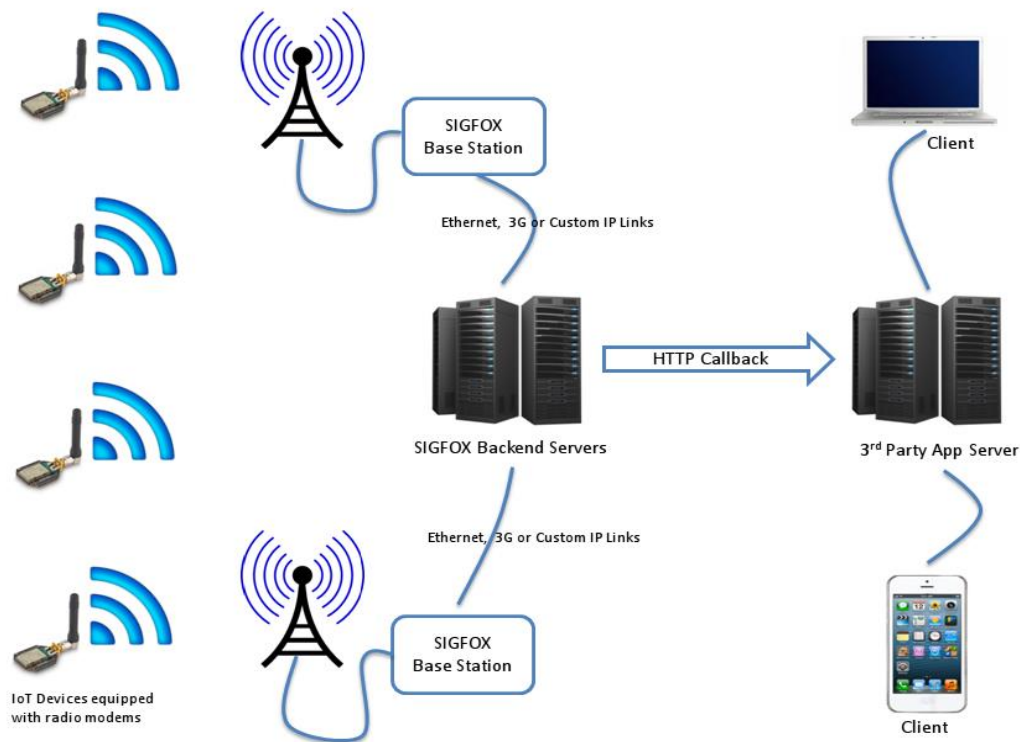


Figure I.8 : Localiser un camion avec GPS à l'aide de GSM réseau cellulaire

### I.5.3.2. Réseaux radio bas-débit

**Sigfox** : sigfox est un pionnier de la communication bas débit (machine to machine) M2M avec connectivité cellulaire l'objectif de cet opérateur de télécommunications est de réduire la consommation d'énergie utilisée par les appareils connectés et leur prix, Sigfox a créé un réseau longue portée de 10 kilomètres en milieu urbain et 30 à 50 km en milieu rural et à bas débit qui permet une communication de données à taille réduite entre les appareils connectés sans passer par un téléphone mobile. Cette connexion bas débit entre machines connectées est possible grâce à la technologie radio Ultra étroite bande (UNB). À faible consommation d'énergie, il utilise des bandes de fréquences libres de droits accessibles à tous, telles que les bandes ISM. De plus, Sigfox a une devise prometteuse qui se réalise jour après jour : « Un réseau, un milliard de rêves ». Le but, pouvoir connecter le monde réel avec le monde virtuel [9].



*Figure I.9 : Architecture d'un réseau SIGFOX*

**LoRaWAN :** La beauté de cette technologie est qu'elle est basée sur une norme ouverte. Elle utilise le spectre sans licence dans le cadre de la bande radio ISM (Industrial, Scientific and Medical). En Europe, LoRaWAN utilise le plan 868 MHz, alors qu'aux États-Unis, il s'agit du plan 915 MHz. L'utilisation du spectre sans licence permet à quiconque de mettre en place son propre réseau et de l'utiliser. De nombreux opérateurs de télécommunications ont commencé à adopter la technologie LoRaWAN et fournissent une connectivité et des services dans de nombreux pays du monde. KPN, Orange, SK Telecom, Comcast et bien d'autres participent activement à des déploiements à grande échelle dans leur région. Cela rend la technologie LoRaWAN encore plus intéressante, car elle est compatible avec les réseaux déployés par différents opérateurs, de la petite à la grande échelle.

La norme LoRaWAN est régie par LoRa Alliance, un groupe de plus de 500 membres qui soutiennent tous les protocoles et fabriquent des composants, des produits et des services liés à LoRa WAN. Des sociétés comme Microchip, ST, Cisco, Softbank et ARM en sont membres. Il permettent de transmettre des données en petite quantité, à des distances de 1 kilomètre en zone urbaine et jusqu'à 20 kilomètres en zone rurale.

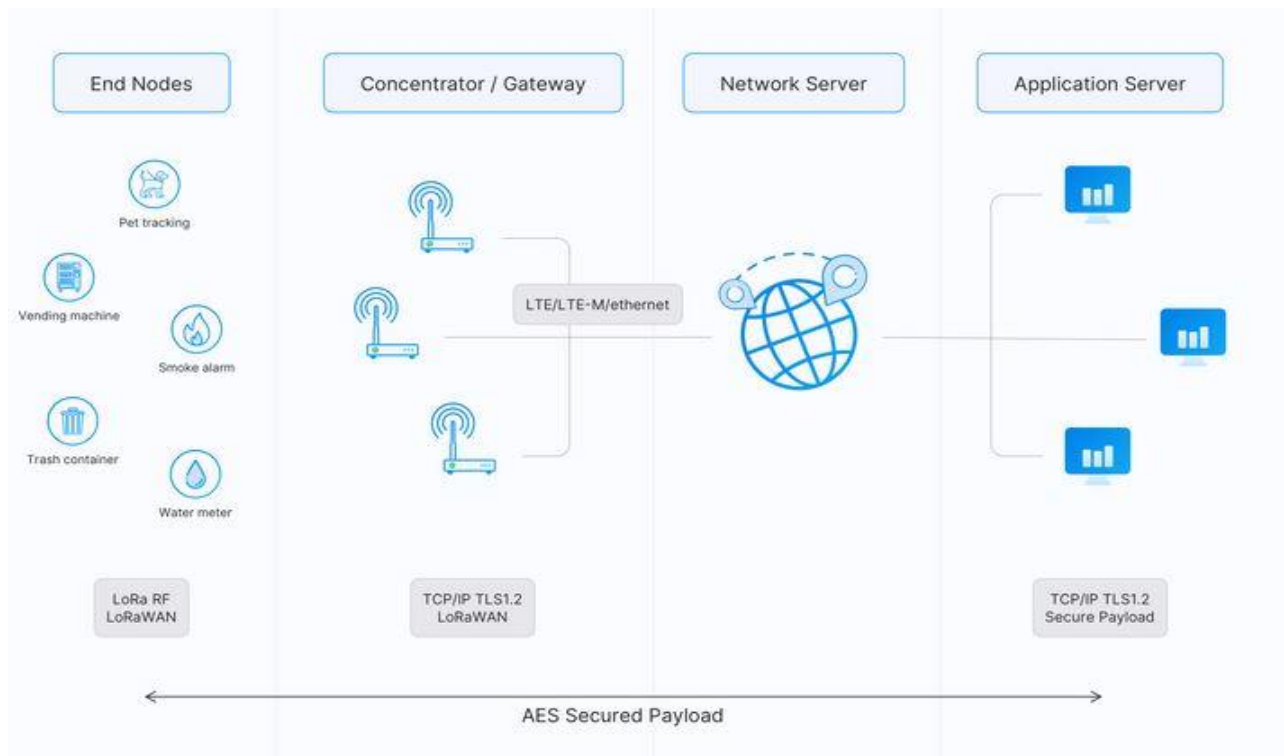


Figure I.10 : Une architecture de réseau LoRaWAN typique

### I.5.3.3. Réseaux propriétaires

Certaines grandes entreprises préfèrent d'installer leurs propres réseaux. Ces réseaux restent privés et sont transparents pour l'utilisateur de réseaux Internet. Ils permettent en fait, par exemple pour une entreprise, de s'affranchir de certaines contraintes, telles que la localisation géographique. Ils rendent possible une transmission plus sécuritaire des données et se fondent également sur un protocole radio à longue portée spécifique et propriétaire : une bande de fréquences radio réservée (169 MHz) est utilisée pour assurer la communication des données entre les compteurs et les concentrateurs de données, comme par exemple, Orange a choisi d'installer son propre réseau de communication et plusieurs Company en prend ce chemin.

## I.6. Domaines d'application de l'Internet des Objets

On peut dire qu'Internet peut être connecté à n'importe quoi. En conséquence, il existe d'innombrables applications pour l'IoT, on cite I.11 quelques applications.

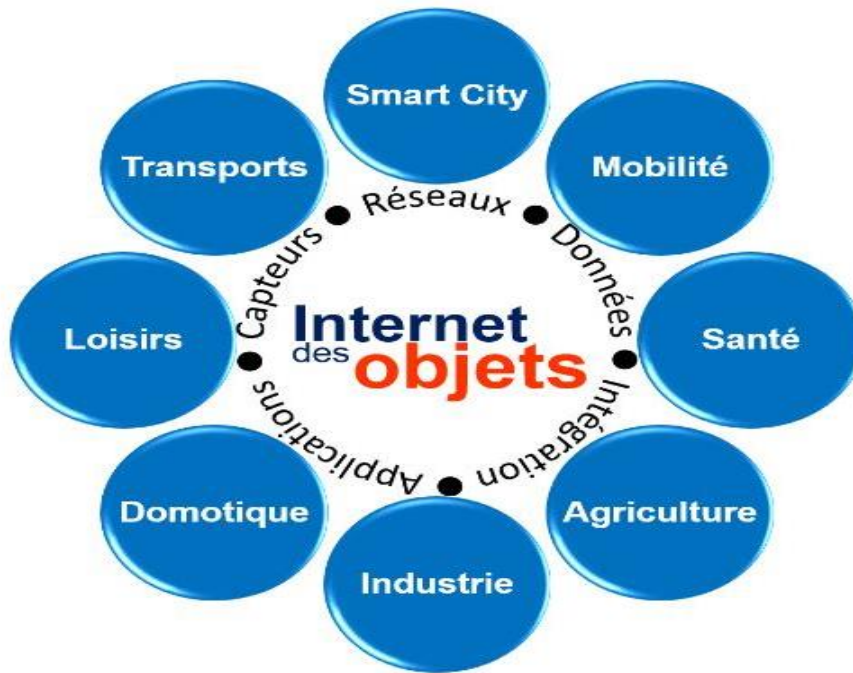


Figure I.11 : Domaines d'application de l'Internet des objets

### I.6.1. Ville intelligente

Qui dit Smart City dit solutions technologiques mises au service des citoyens plus fournir des nouveaux services pour améliorer la vie publique des gens. Une ville connectée doit se construire à partir des besoins de ses citoyens, et non pas à partir des technologies disponibles.

La ville intelligente pourrait se décomposer en trois couches technologiques, qui permettent une meilleure administration des situations et améliorer la qualité de vie des usagers :

Une première couche est constituée d'infrastructures : ce sont les réseaux d'eau, d'énergie ou de télécommunication, le mobilier urbain ou encore les capteurs connectés, ce niveau et le niveau de base de chaque ville intelligente sans oublier de tout relier par des capteurs et des réseaux de collecte bas débit LoRa, Sigfox. L'utilisation des smartphones et des réseaux 4G sont les fers de lance de cette idée smart city.

La deuxième couche est constituée du système d'information des données collectées et du logiciel de gestion les métiers de la ville. Grâce à cette couche, le logiciel analyse les données collectées en temps réel et permet de trouver des solutions à des problèmes spécifiques. De nombreux outils éprouvés existent actuellement et présentent des résultats à des niveaux industriels :

- Le BIM (Building Information Modeling) est un système dédié à la collecte systématique des données du bâtiment, une des technologies rapproche à la domotique, il permet le suivi des bâtiments à distance;

- L'automatisation, avec l'essor de logiciels de gestion automatisée permettant la mise en œuvre en temps réel de tâches répétitives nécessitant l'analyse de grande masse de données ;
- Le Big Data et le développement de l'Intelligence Artificielle, permettant la maintenance prédictive des équipements et l'identification automatique de dysfonctionnements.

Une dernière couche elle est réservée pour les solutions finales distinguées aux citoyens. À ce titre, de nombreux dispositifs existent d'ores et déjà, telles que :

- En interne : les logiciels de Gestion de Maintenance Assistée par Ordinateur (GMAO), de Gestion Techniques des Interventions (GTI), de supervision métier, de contrôle des achats, d'administration financière, de gestion des ressources humaines...
- Vers les administrés : applications de signalements citoyens (Tell my City, Neocity...), applications citoyennes (Ma ville dans ma poche), places de marché en ligne, application de Gestion de la Relation Usager (GRU) [10].

### LES SIX PILIERS D'UNE SMART CITY

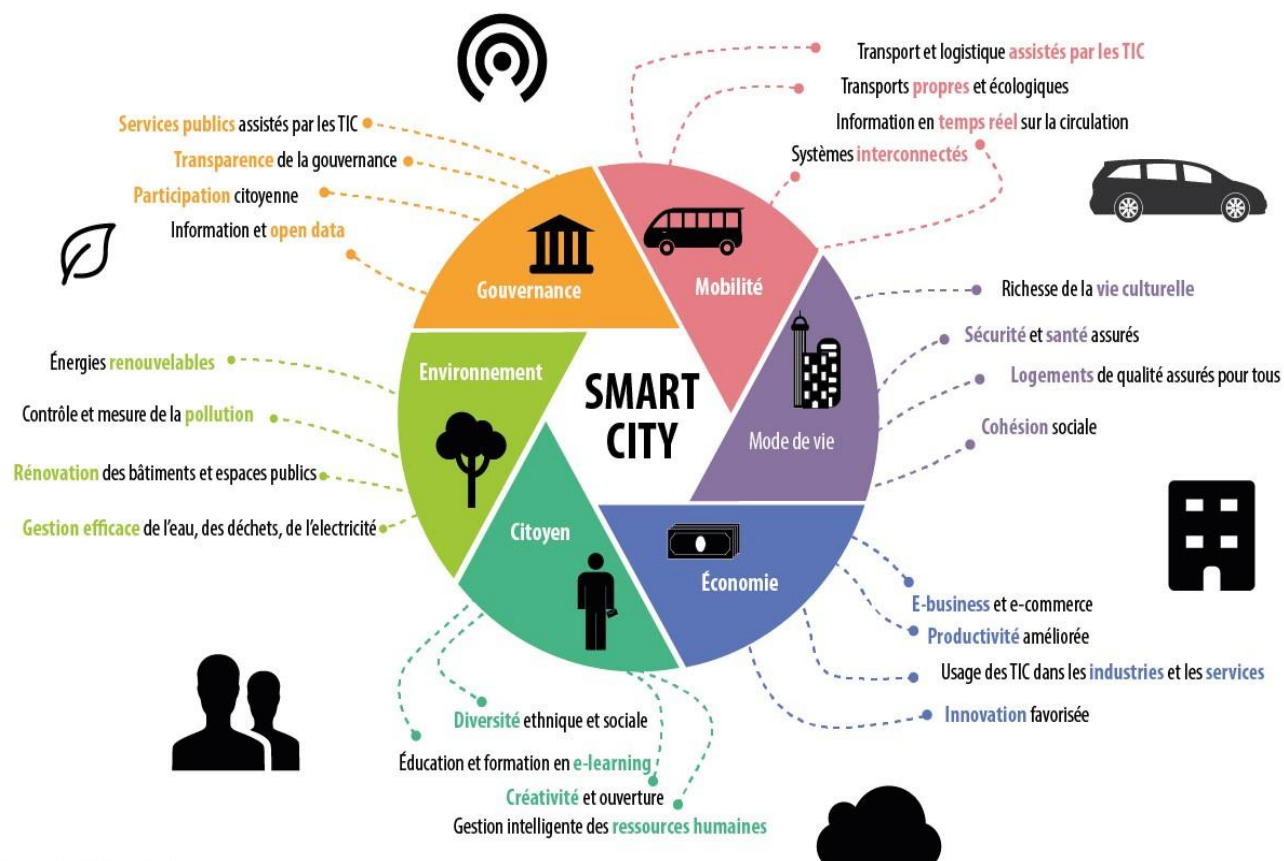


Figure I.12 : Smart City



### I.6.2. La domotique

La domotique regroupe différentes technologies pour permettre de contrôler, programmer et automatiser votre maison. Elle rassemble et utilise les domaines de l'électronique, de l'informatique, des télécommunications et de l'automatique. La domotique opère dans des domaines techniques et informatiques très larges. Elle peut être utilisée pour programmer la plupart des appareils et appareils de la maison, de l'éclairage et du chauffage aux équipements et appareils audiovisuels, y compris l'ouverture des fenêtres. Elle facilite également le contrôle de la maison en gérant les systèmes d'alarme, la protection incendie et même la température ambiante.

Sa fonction est de programmer, contrôler et automatiser à distance ou localement tous les appareils domestiques intégrés au réseau. Ce dernier fonctionne avec ou sans fil pour recevoir et transmettre des données entre les différents points de contrôle et les appareils à contrôler. En sens inverse, chaque appareil peut communiquer son état de fonctionnement au point de contrôle.



*Figure I.13 : Maison intelligente*

### I.6.3. Agriculture intelligente

L'agriculture intelligente vise à renforcer les capacités des systèmes agricoles et à promouvoir la sécurité alimentaire en intégrant les besoins d'adaptation et les capacités d'atténuation dans les stratégies de développement agricole durable. Cet objectif nécessite l'utilisation de nouvelles technologies telles que l'imagerie satellitaire et l'informatique. Elle s'appuie sur des capteurs pour détecter l'environnement et fournir des informations utiles aux agriculteurs sur les conditions du sol, des niveaux d'humidité, etc. L'agriculteur utilisera ces informations pour assurer une production excellente.



Figure I.14 : Exemple de Agriculture intelligente

#### I.6.4. Industrie intelligente

L'internet des objets a changé tout le monde, les smartphones, la domotique et l'industrie ne font pas exception, il devient l'un des principes de ce domaine sous le nom d'industrie (4.0), les équipements industriels sont tous mis en réseau pour devenir interconnectés. Grâce à des capteurs, des détecteurs, des machines, des opérateurs collectant des données de plus en plus géantes et dont la gestion, l'analyse et l'exploitation deviennent un jeu colossal pour les entreprises.

Lorsque l'internet des objets est né, l'industrie a commencé à évoluer vers l'industrie 4.0, porteuse de promesses en matière de productivité, de qualité, de performance et de sécurité.

En faisant évoluer les processus de fabrication en ajoutant des capteurs et surtout en connectant les équipements entre eux et avec le système d'information de l'entreprise qui l'est.

Cette évolution permet de réduire les problèmes, puisque les capteurs connectés et les détecteurs enregistrent les données relatives à la défaillance de manière plus précise. Il devient donc plus facile d'identifier la source de non-conformité, puis d'y remédier.

Enfin, l'internet des objets contribue à améliorer la qualité de l'industrie grâce à un suivi plus efficace des différentes étapes de production.

L'internet des objets offre une visibilité générale en temps réel des états des machines donc les appareils IoT sont dédiés à la maintenance, et à la maintenance prédictive en particulier.

Grâce à l'analyse des données pendant le fonctionnement de la machine (nombre d'heures, tours, températures, vibrations, etc.) des alertes sont émises pour prévenir d'une panne à venir, l'exécution et les interventions de maintenance deviennent plus efficaces.



Les wearables offrent aux patients à risque de troubles cardiovasculaires une mesure des battements cardiaques pour s'assurer qu'une reprise d'activité physique donne de bons résultats sur le rythme cardiaque au repos.

Peut également utiliser la santé intelligente pour analyser les habitudes et les différentes phases de sommeil d'une personne souffrant de troubles du sommeil afin d'ajuster ses heures de lever et de coucher.

Aussi en Peut analyser la composition de tous les aliments par un scanner connecté à un smartphone Pour les allergies et intolérances alimentaires.

### I.6.5.3. Les applications / objets connectés dans l'exercice médical

Selon la CNS (Conférence Nationale de la Santé) les objets connectés sont un élément essentiel pour l'évolution de la médecine 4p (préventive, prédictive, personnalisée, participative) mais aussi ils portent des effets néfastes sur l'organisation du système de santé et du système de solidarité et ils demande d'ouvrir un débat, pour mieux encadrer l'usage et la recommandation de ces produits et services, elle insiste aussi sur le fait que ces objets doivent rester des outils et non une compétence à part entière [12].

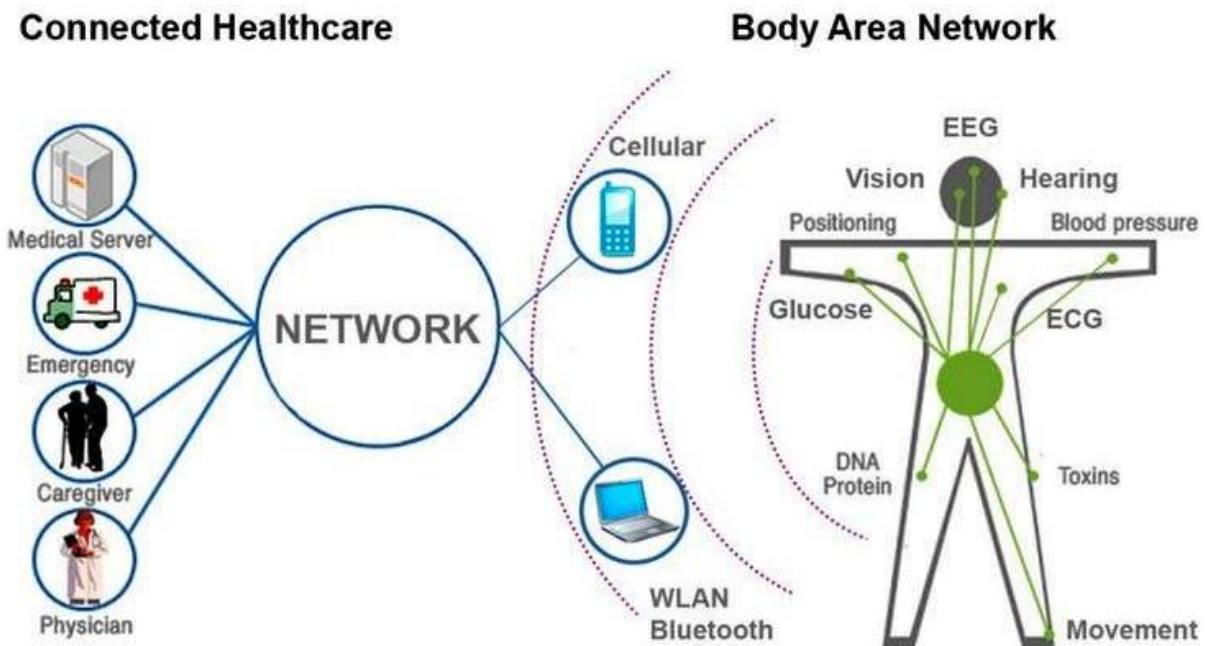


Figure I.16 : Santé intelligente

## I.7. Les objets intelligents dans l'IoT

En effet, les données sont une véritable mine d'or pour l'IoT. Dans la mesure où elle est captée et transformée en donnée intelligente bien évidemment. L'analyse d'une donnée brute, de température, de vibration, d'humidité... a pour but de la rendre exploitable. C'est dans cette optique que les objets connectés sont :

- Instrumentés : ils collectent et intègrent les données en temps réel.

- Interconnectés : elles sont partagées à l'aide de réseau sans fil à d'autres systèmes d'information.
- Intelligents : elles sont analysées et enrichies pour aider les usagers, les entreprises et les institutions dans leurs prises de décisions [13].

### **I.8. L'importance des objets connectés dans la santé**

- Pourquoi la santé connectée est-elle importante ?

Pour un système de santé plus développé, il doit être une combinaison entre la prestation de soins de santé et l'innovation technologique et de systèmes d'information pour avoir une meilleure qualité de soins grâce à une bonne gestion de l'information et à la collaboration entre les hôpitaux et les centres de santé et les médecins.

Pour mieux sécuriser les personnes fragiles et les personnes âgées qui vivent seules ou en Ephad, il faut que la téléassistance soit utilisée, ce nouveau système permet d'évoluer et de renforcer la collaboration et la qualité des soins en santé, aussi permet un transfert transparent des données dans l'échange entre le médecin et le patient ou entre deux médecins.

La santé intelligente utilise les technologies de l'information pour fournir des services de santé de manière plus efficace et efficiente, elle permet la collaboration et l'échange d'informations au niveau interdisciplinaire.

- Comment la santé connectée facilite-t-elle la prévention en médecine ?

La santé connectée a minimisé le temps de diagnostic du patient et apporte une meilleure efficacité en mettant en relation chaque patient par son médecin qui interagit avec le patient à tout moment sans oublier, le médecin soit informé de tous les détails du patient et de sa condition physique, cela permet de donner un traitement rapide et des recommandations si nécessaires.

Avant cette opération n'est pas possible avec la méthode traditionnelle car chaque visite nécessite une demande d'information séparée auprès de différents services.

- Quel est le rôle des objets connectés dans la santé connectée ?

Le monde a trop profité de la santé intelligente, la télésurveillance dans le secteur médical permet aux médecins de proposer des soins aux patients à tout moment et surtout de prévenir des dégâts comme une crise cardiaque, cette technologie offre une grande sécurité pour tous les malades. Sans oublier à faire évoluer ce domaine en utilisant des interactions entre professionnels de santé qui devinent plus facilement.

Les appareils connectés offrent beaucoup de solutions au domaine de la santé ils changent complètement la façon de diagnostiquer les patients et son grand avantage est d'être alerté en cas d'incident.

Téléphones portables, montres, bracelets, balances, T-shirts, chaussures, etc., tous ces objets sont aujourd'hui connectés ! Ils permettent aux patients d'être mieux informés, et contrôlés, et favorisent ainsi la prévention [13].

## **I.9. Les enjeux de l'Internet des Objets**

Un défi majeur pour l'IoT à la fois technique et utile, c'est de pouvoir gérer l'hétérogénéité technologique et des normes d'objets couplées à des multitudes de besoins d'applications et des usages en terme de services de sécurité. Sachant que ces besoins peuvent évoluer dans le temps selon le contexte. En effet, comment assurer l'authentification individuelle de plusieurs millions d'objets hétérogènes, dotés des technologies de communication hétérogènes, à travers des domaines administratifs multiples ? Ceci soulève des problèmes de gestion et de sécurité des objets dans un milieu hétérogène au niveau physique et logique. Il est donc important d'instaurer des politiques de sécurité claires, adaptatives selon le contexte d'utilisation et d'établir les responsabilités des faits et des actions sur l'environnement physique des objets. « En effet, l'Internet des objets est un système complexe dans lequel interagissent des personnes avec un écosystème technologique à base d'objets intelligents à travers des processus complexes. Les interactions de ces quatre composantes de l'IoT (personnes, objets intelligents, écosystème technologique, processus) font émerger une dimension systémique à la sécurité de l'IoT ». Par la suite, les tensions sur la sécurité de l'IoT sont créées lors de l'interaction des objets intelligents avec son environnement [14].

### **I.9.1. Risques d'utilisation des objets connectés**

Les objets connectés stockent souvent des données sensibles, accessibles et transmises en continu. Les systèmes de sécurité tels que les caméras et les sonnettes font de plus en plus partie des réseaux et peuvent rapidement causer des problèmes majeurs s'ils sont attaqués par des cybercriminels. Les équipements de bureau tels que les imprimantes sont également des points d'accès potentiels : si l'imprimante est piratée, le pirate peut voir tout ce qui est imprimé ou numérisé au bureau. Le piratage de l'objet connecté permet aux pirates d'accéder à toutes ses fonctions. Le piratage des systèmes de chauffage ou des machines ou d'un appareil de santé peut être un problème plus important pour les gens. Des individus malveillants peuvent prendre en otages des véhicules et leurs passagers ou exiger un paiement pour mettre fin au sabotage de la chaîne de montage, par exemple, une attaque DDoS (Distributed Denial of Service) envoie un flux de requêtes réseau ciblées d'un appareil infecté à un serveur, un ordinateur ou un réseau que les pirates veulent compromettre. Étant donné que la cible doit gérer trop de demandes réseau, elle se bloque et devient indisponible pour les utilisateurs réels.

### **I.9.2. Sécurisation de l'appareil**

Tout système IoT il peut être piraté surtout en cas de développement ou lors de processus de recherche c'est pour sa la fourniture de systèmes de sécurité il est indispensable comme les systèmes d'exploitation récents.

Les développeurs de systèmes IoT doivent être conscients aux vulnérabilités de la cybersécurité à chaque étape du développement, et pas seulement lors de la

---

phase de conception. Par exemple, le piratage d'un système de voiture intelligent peut causer des gros dégâts

### **I.9.2.1. PKI et certificats numériques CA**

La meilleure façon de sécuriser la discussion entre le serveur et le client avec un énorme appareil dans le réseau est d'utiliser la cryptographie asymétrique (clé privée, clé publique) pour chaque utilisateur afin de faciliter le cryptage et le décryptage des données à l'aide de certificats numériques CA, ce système rend possible de sécuriser les informations saisies sur le site pour effectuer des transactions privées, l'achat sur internet ne peut fonctionner sans PKI.

### **I.9.2.2. Sécurité des réseaux**

Les réseaux représentent le cœur de la communication entre les objets connectés et le contrôle à distance, il joue un rôle énorme pour les utilisateurs d'objets connectés, ils contiennent des composants numériques et physiques La sécurité IoT doit englober les deux composants, la protection d'un réseau IoT comprend assurer la sécurité des ports, désactiver le transfert, l'ouverture du port sauf si nécessaire, l'utilisation d'anti-malware, de pare-feu et de systèmes de détection et de prévention des intrusions, le blocage des adresses IP non autorisées et la garantie que les systèmes sont corrigés et à jour.

### **I.9.2.3. Sécurité des API**

La sécurité des API est très importante pour protéger la confidentialité et l'intégrité des données envoyées par les appareils connectés aux systèmes dorsaux. Les API sont l'épine dorsale de la plupart des sites Web sophistiqués, il faut donc s'assurer qu'à l'exception des appareils, des développeurs et des applications autorisées communiquent avec les API. En 2018, T-mobile, le géant de la téléphonie mobile, a perdu près de 2 millions des données clients, ainsi que leurs codes postaux, numéros de téléphone et numéros de compte en raison d'une mauvaise gestion de la sécurité des API et c'est un excellent exemple de mauvaise gestion des données.

Enfin, avec le développement rapide des objets connectés, le risque de piratage de données augmente de plus en plus et devient un véritable problème pour les fabricants pour cela, leur développeur doit renforcer les protocoles de sécurité et également responsabiliser l'utilisateur afin qu'il connaisse les risques de passer par de tels objets, plus donner des conseils à l'utilisateur tels que la mise à jour du firmware, la sécurisation de votre réseau, la modification des mots chaque mois [15].

### **I.9.3. Types d'environnement cloud**

Il existe trois types d'environnements cloud : public, privé et hybride. Chaque plate-forme cloud sert le même objectif. Il partage les ressources informatiques sur le réseau et permet la fourniture de services basés sur le cloud.

### **I.9.4. Sécurisation du cloud**

La sécurité cloud signifie protéger tous les éléments d'un environnement cloud, y compris l'infrastructure, les données et les applications. Chaque fournisseur de cloud public propose des ensembles de contrôles de sécurité différents. Les paramètres de configuration par défaut diffèrent aussi selon le fournisseur. En dépit de certaines correspondances, on relève des mises en œuvre différentes et des nuances dans le déploiement. Tant que les entreprises ne seront pas capables de sécuriser l'ensemble de leurs cloud, les cybers adversaires continueront de tirer parti des erreurs de configuration.

#### **I.9.4.1. Solutions CrowdStrike de sécurité du cloud**

Bloquer toutes les intrusions et l'espionnage des données dans le cloud nécessite une plate-forme étroitement intégrée est essentielle chaque fonction en bas joue un rôle important dans la détection des menaces et doit être conçue et développée avec une attention constante à la vitesse, à l'évolutivité et à la fiabilité.

La sécurité du cloud moderne va plus loin que les approches ad hoc, en unifiant tous les outils requis en une seule plateforme :

Falcon Horizon (CSPM) : permet l'identification et la correction de risques au niveau des infrastructures cloud, dont IaaS, SaaS et PaaS.

Falcon Cloud Workload Protection : il est dédié aux menaces avancées et offre une visibilité tout au long de DevOps et une intégration/livraison continue, il offre une infrastructure stable qui optimise les ressources cloud et surveille en permanence la sécurité des applications.

Sécurité des conteneurs avec Falcon : La plateforme Falcon protège en continu les conteneurs contre les vulnérabilités. Cette couverture comprend le pipeline des conteneurs, l'infrastructure de déploiement et la supply chain [16].

### **I.9.5. Gestion du cycle de vie d'un objet connecté**

Le cycle de vie d'un objet connecté est divisé en trois phases : le «Bootstrapping et l'enregistrement», la «Découverte» et finalement la phase «Opération». Chacune de ces étapes doit garantir la sécurité et la protection des données privées des entités impliquées (objet / utilisateur). Les informations et les ressources de l'objet doivent être également protégées. Ces trois étapes supposent l'existence d'une entité particulière habituellement appelée fournisseur d'objet, cumulant les rôles de serveur et de base de données.

Lors de la première étape de Bootstrapping et d'enregistrement, l'objet doit d'abord être installé et configuré. Son authentification auprès du fournisseur

---



survient ensuite accompagnée d'une autorisation de déploiement dans le réseau. Ces opérations doivent être effectuées via un protocole compatible avec les contraintes liées à la capacité de calcul et à la consommation énergétique des objets. Plusieurs solutions ont été ainsi proposées comme HIP-DEX, PANA, EAP, et 802.1x. Cette étape peut aussi être utilisée pour calculer par exemple une identité partielle qui pourra servir à anonymiser des échanges ultérieurs. La réalisation de l'enregistrement de l'objet auprès du fournisseur, permet d'amorcer la seconde phase.

L'étape de Découverte permet à une entité de l'IoT de déterminer, via une étape de localisation préalable, les objets disposant des ressources dont elle a besoin. L'entité demandeuse doit alors s'adresser au fournisseur en utilisant un protocole comme LDAP, HS, ... Toutes ces solutions nécessitent l'authentification et l'autorisation de l'entité demandeuse.

La dernière étape est l'Opération, où un objet A essayé de communiquer avec un objet B, tout en sécurisant cette communication. La figure I.17 donne un aperçu des différents messages échangés entre les deux objets afin de créer un canal sécurisé de communication. Elle regroupe l'étape de découverte (B veut accéder à certaines ressources possédées par A et éventuellement par d'autres entités) et l'opération [17].

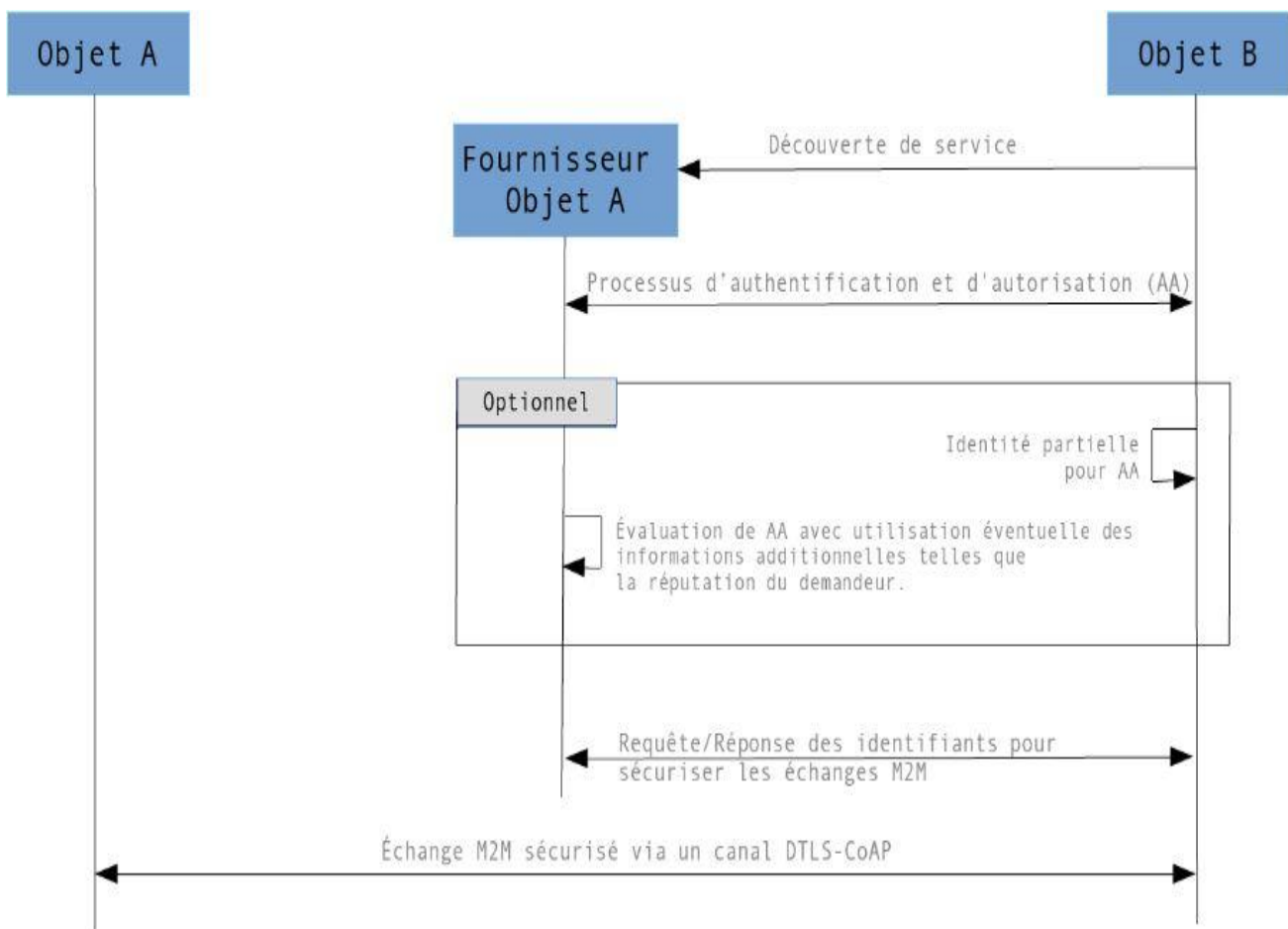


Figure I.17 : Cycle de vie d'un objet connecté

### I.10. Le Web des objets

Le Web des objets a permis aux développeurs Web de trouver plus facilement de l'aide pour créer un système IoT avec juste une application Web parmi les exemples trouve une API de script, ainsi que le binding des Templates.

- Quelle est l'architecture du Web of Things ?

La normalisation de WoT (Web of Things) par le W3C repose sur plusieurs blocs de construction. Il s'agit de :

- La description de l'objet « Thing ».
- Le binding des protocoles.
- L'API de Scripting.
- Les directives de sécurité et de confidentialité [18].

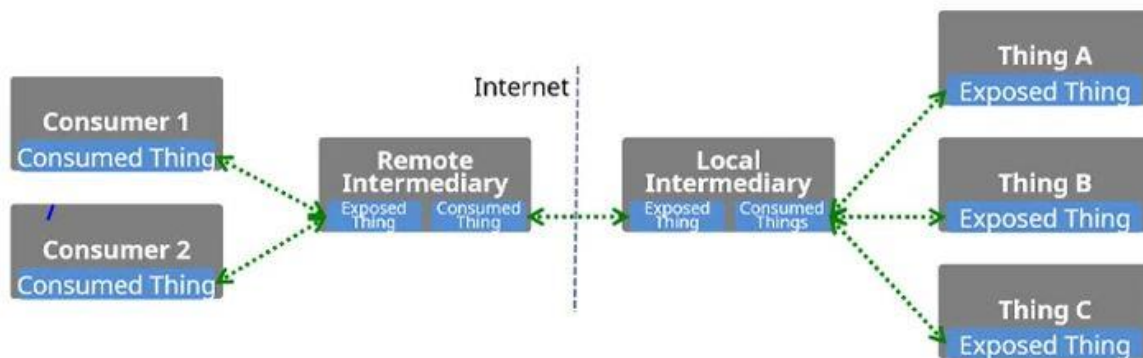


Figure I.18 : L'architecture du Web of Things (WoT)

### I.11. Conclusion

Aujourd'hui, les technologies les plus couramment utilisées peuvent se connecter à Internet. Les actifs peuvent alors fournir des données plus nombreuses et plus riches. Les informations enregistrées par ces compteurs, capteurs et autres appareils connectés peuvent guider les processus de prise de décision en temps réel et soutenir la création des systèmes intelligents dans n'importe quel domaine ; Dans le chapitre suivant, nous allons présenter le système IoT réalisé.

## *Chapitre II :*

*Présentation du système IoT réalisé*

## **II.1. Introduction**

Dans ce chapitre, nous allons décrire notre système de surveillance. C'est un système d'objets connectés pour le suivi des patients cardiaques.

Puis, nous allons mettre en évidence l'environnement globale de réalisation c'est à dire l'ensemble matériels et logiciels nécessaires pour le développement.

## **II.2. Présentation du système**

Comme on a vu dans le chapitre précédent, un modèle d'Internet des objets est constitué de trois principales couches. Notre prototype proposé suit cette même décomposition.

La première couche est équipée d'une carte WEMOS D1 Mini (ESP8266) permettant d'effectuer les calculs nécessaires et en même temps de se connecter au réseau WiFi afin de transmettre les données collectées par le système à la base de données en temps réel, un module GSM/GPRS/GPS (SIM808) doté d'une antenne GPS pour détecter la position GPS d'un patient, d'une antenne GSM pour envoyer une alerte SMS en cas d'anomalies, et d'un capteur de fréquence cardiaque (AD8232) pour capter le signal ECG.

Pour la deuxième couche, on va se servir d'un point d'accès sans fil (WiFi ) pour connecter à l'Internet afin d'envoyer les informations collectées par le capteur, et une carte SIM pour s'attacher aux réseaux GSM afin d'envoyer des alertes par SMS.

Dans la troisième couche, on a utilisé des plateformes IoT Cloud pour recevoir les informations envoyées par notre système, de plus, on a développé une application Web.

Le principe de fonctionnement de ce système est basé sur l'acquisition du signal ECG d'un patient à l'aide du capteur AD8232, compter le nombre de battements par minute et le comparer aux valeurs normales pour détecter et diagnostiquer les anomalies cardiaques, puis les envoyer en temps réel à la base de données via la connexion WiFi à travers le WEMOS. D'autre part, en cas d'anomalies le système doit détecter la position GPS et envoyer des alertes SMS au médecin à l'aide de SIM808.

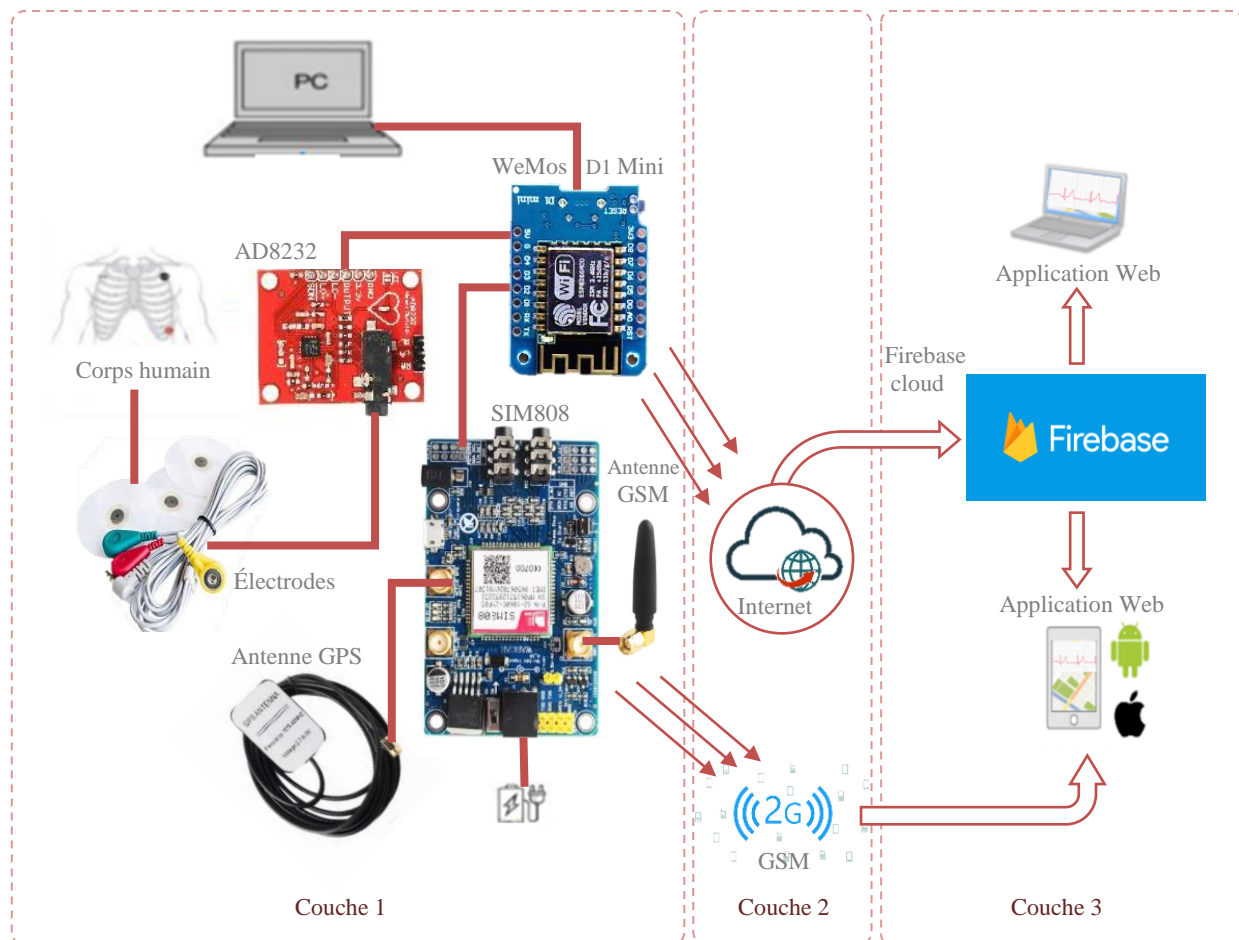


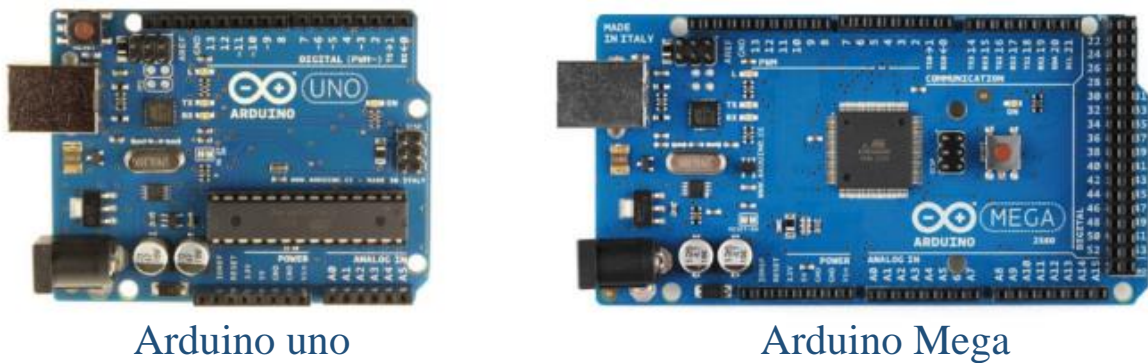
Figure II.1 : Conception globale du système proposé

## II.3. Matériel requis

### II.3.1. La carte Arduino

#### II.3.1.1. Définition

Le système Arduino est une plateforme matérielle et logicielle de développement d'applications embarquées. C'est un circuit imprimé open-source d'électronique programmée basée sur une simple carte à microcontrôleur (de la famille AVR) comportant un certain nombre d'entrées et de sorties (les ports) permettant la connexion de capteurs, ou d'actionneurs, et un logiciel, véritable environnement du développement intégré, pour écrire, compiler et transférer le programme vers la carte à microcontrôleur [19].



Arduino uno

Arduino Mega

*Figure II.2 : Exemples du carte Arduino*

### II.3.1.2. Les gammes de la carte Arduino

Actuellement, il existe plus de 20 versions de module Arduino, nous citons l'Arduino UNO, l'Arduino Mini, l'Arduino Nano, l'Arduino Micro, et l'Arduino Mega2560.

*Figure II.3 : Les gammes de la carte Arduino*

Le tableau II.1 représente la chronologie des sorties de cartes Arduino :

Version de carte	Année de sortie	Micro-contrôleur
Diecimila	2007	ATmega168V
LilyPad	2007	ATmega168V/ATmega328V
Nano	2008	ATmega328/ATmega168
Mini	2008	ATmega168
Mini Pro	2008	ATmega328
Duemilanove	2008	ATmega168/ATmega328
Mega	2009	ATmega1280
Fio	2010	ATmega328P
Mega 2560	2010	ATmega2560
Uno	2010	ATmega328P
Ethernet	2011	ATmega328
Mega ADK	2011	ATmega2560
Leonardo	2012	ATmega32U4
Esplora	2012	ATmega32U4
Micro	2012	ATmega32U4
Yún	2013	ATmega32U4 +Linino

*Tableau II.1 : Chronologie des sorties de cartes Arduino [20]*

Lorsque plusieurs types de micro-contrôleurs sont indiqués, cela signifie qu'une première version a été produite avec le premier type et plus tard avec l'autre (généralement plus puissant).

Dans notre projet, nous avons utilisé le module WEMOS D1 Mini (ESP8266) au lieu de la carte Arduino car il nous offre des fonctionnalités supplémentaires comme la connexion aux réseaux WiFi.

### **II.3.2. Le WEMOS D1 Mini (ESP8266)**

#### **II.3.2.1. Définition**

Le module WEMOS D1 Mini est un microcontrôleur avec module WiFi intégré. C'est une carte de développement à faible coût dédiée à l'internet des objets (IoT), très simple à utiliser et possède une capacité de mémoire et de calcul

supérieure aux Arduino. Cette carte peut se connecter à des réseaux WiFi existants comme un client ou un serveur ou bien de mettre en place son propre réseau grâce à un émetteur-récepteur WiFi.

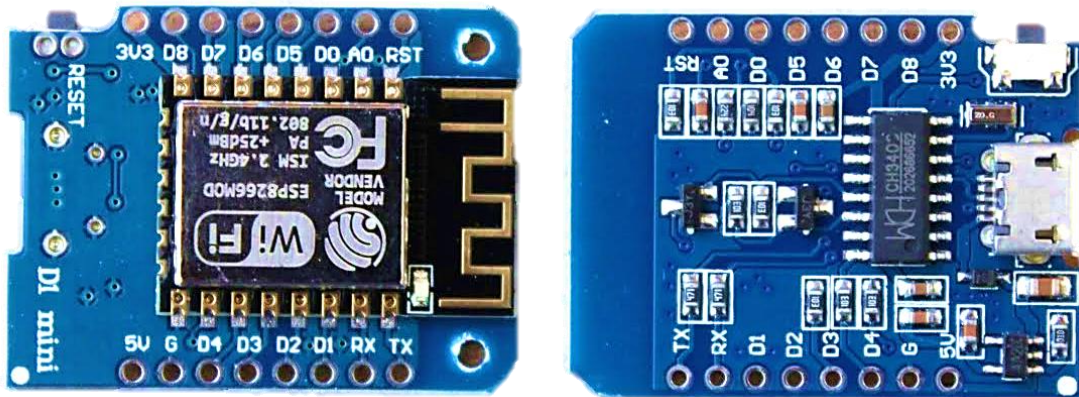


Figure II.4 : Le module WEMOS D1 Mini

### II.3.2.2. Caractéristiques

Parmi les caractéristiques du module WEMOS D1 Mini, on trouve :

Microcontrôleur	ESP8266EX
Tension de fonctionnement	3.3V
Entrées /sorties numériques	11 (3.3V max)
Entrées analogiques	1 (3.2V max)
Mémoire FLASH	4 MB
Mémoire RAM	64 KB
Fréquence d'horloge	80/160 MHz
Interfaces	SPI, I2C, I2S, UART
Interface Wifi	IEEE 802.11 b/g/n
Longueur x Largeur	34,2 x 25,6 mm

Tableau II.2 : Les caractéristiques de la carte WEMOS D1 Mini [21]

### II.3.2.3. Les broches de WEMOS D1 Mini

Le tableau II.3 et la figure II.5 montrent les différentes broches du module WEMOS.



Broche WEMOS	Les fonctions	ESP8266 pin
TX	Donnée série, transmettre	TXD
RX	Donnée série, recevoir	RXD
A0	Entrée analogique	A0
D0	E/S numérique	GPIO16
D1	E/S numérique, SCL	GPIO5
D2	E/S numérique, SDA	GPIO4
D3	E/S numérique, pull-up 10k	GPIO0
D4	E/S numérique, tractions 10k, LED	GPIO2
D5	E/S numérique, SCK	GPIO14
D6	E/S numérique, MISO	GPIO12
D7	E/S numérique, MOSI	GPIO13
D8	E/S numérique, Pull-down 10k, SS	GPIO15
G	La masse	GND
5V	5V	
3V3	3,3 V	3,3 V
RST	Réinitialiser	RST

Tableau II.3 : Les broches de module WEMOS D1 Mini [21]

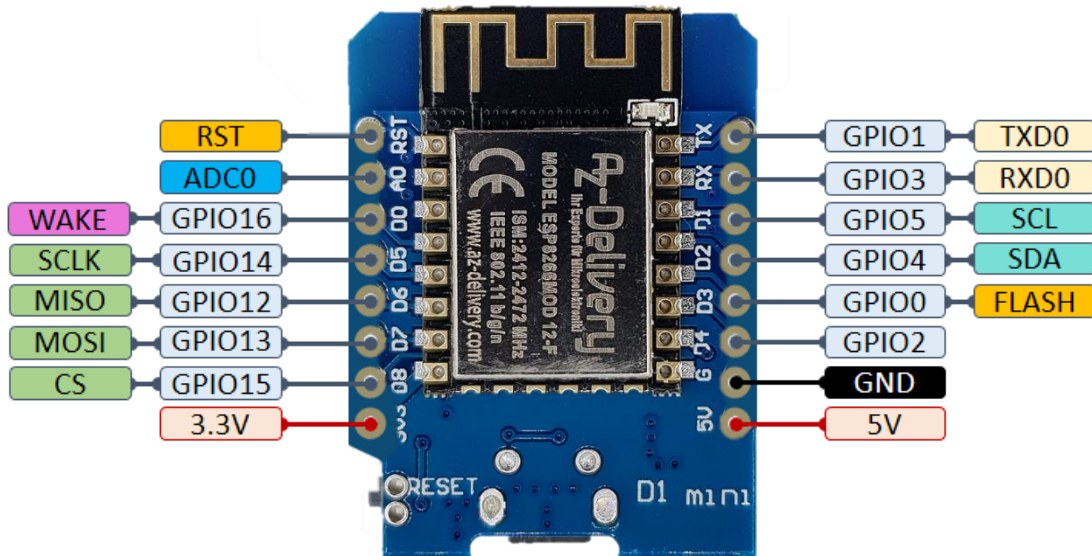


Figure II.5 : Les broches de module WEMOS D1 Mini

### II.3.3. Le module GSM/GPRS/GPS (SIM808)

#### II.3.3.1. Définition

Le module SIM808 est un composant compact développée par SIMCOM. Il s'agit de l'intégration de GSM/GPRS et de GPS qui prend en charge les fréquences quadri-bande de 850/900/1800/1900 MHz et fusionne avec la technologie GPS pour obtenir la navigation par satellite avec 22 canaux de suivi et 66 canaux récepteurs d'acquisition. Ce module réduit le temps, l'énergie et l'argent nécessaires à la création d'applications basées sur le GPS [22].

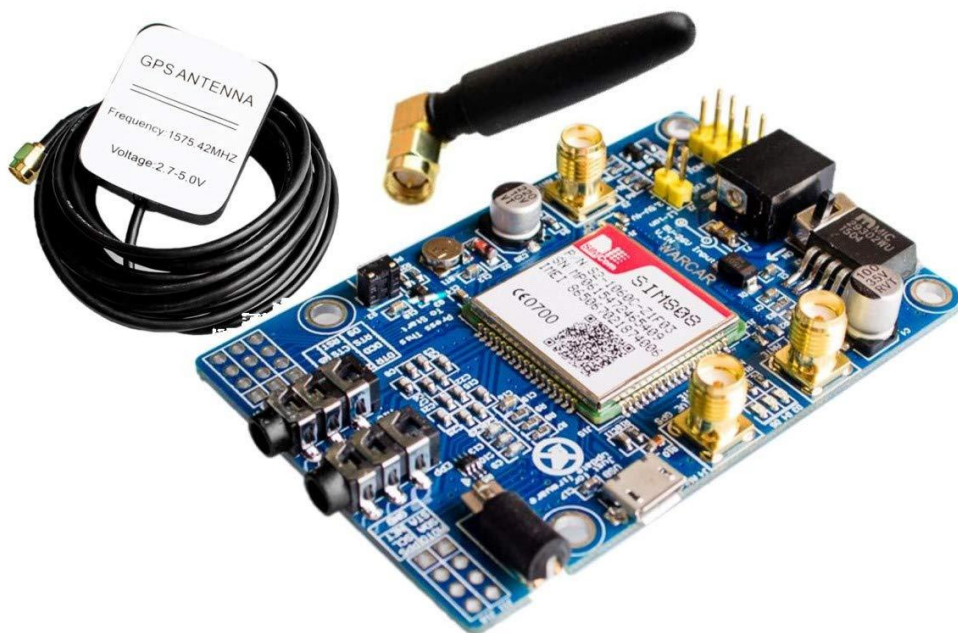


Figure II.6 : Le module GSM/GPRS/GPS (SIM808)

Le SIM808 est contrôlé par des commandes AT et prend en charge 3,3V et 5V. Il se connecte directement à l'antenne GSM et GPS par un connecteur d'antenne externe.

Le module GSM/GPRS/GPS (SIM808) offre les fonctions suivantes :

- Envoyer et recevoir des données GPRS (TCP / IP, HTTP, etc.).
- Recevoir des données GPS et des données A-GPS.
- Envoyer et recevoir des SMS.
- Faire et recevoir des appels téléphoniques.

#### II.3.3.2. Interfaces

- Interface audio analogique
- Interface PCM (optionnel)
- Interface SPI (optionnel)
- Sauvegarde RTC

- Interface série
- Interface USB
- Interface vers SIM externe 3V / 1.8V
- Interface clavier
- GPIO
- ADC
- Entré d'antenne GSM
- Entré d'antenne GPS
- Entré d'antenne Bluetooth

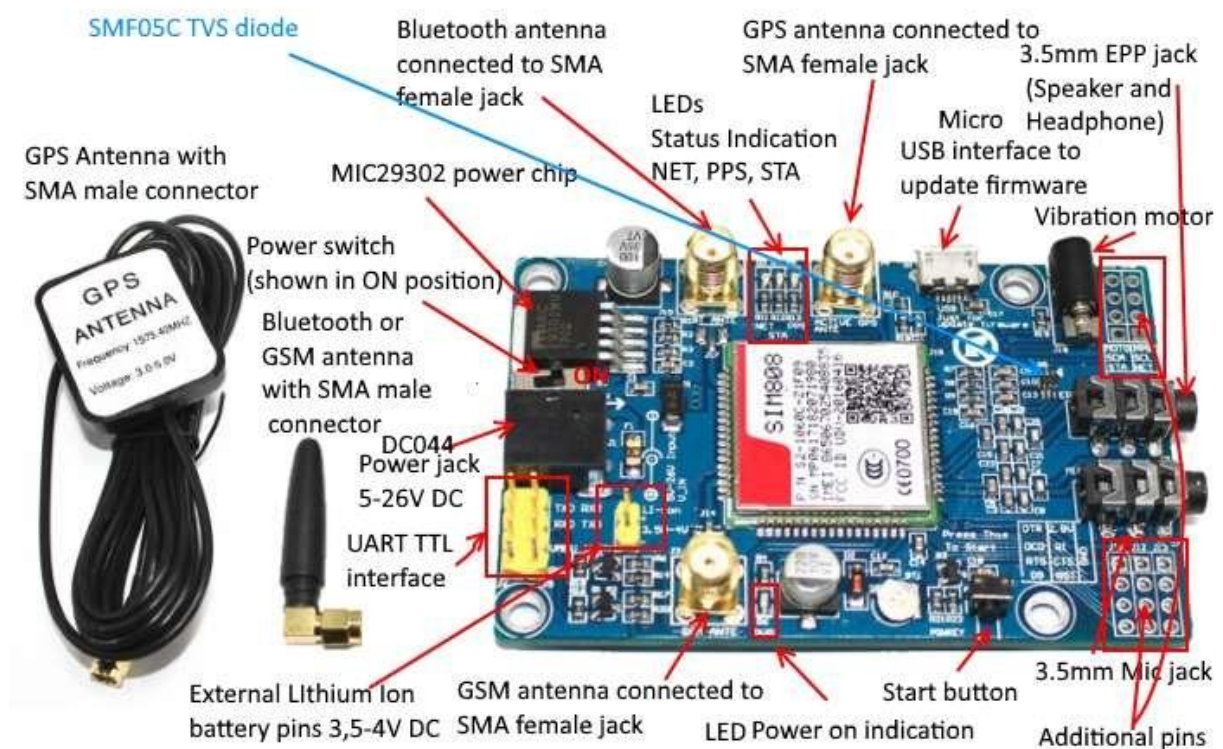


Figure II.7 : Interfaces du SIM808

### II.3.3.3. Caractéristiques générales

Parmi les caractéristiques générales de module SIM808, on peut citer les suivantes [6] :

- Tension de fonctionnement 5V
- Quadri-bande 850/900/1800 / 1900MHz
- GPRS multi-slots classe 12/10
- Station mobile GPRS classe B
- Conforme à la phase GSM 2/2 + (classe 4 (2 W à 850/900 MHz; classe 1 (1 W à 1800/1900 MHz))
- Bluetooth: compatible avec 3.0 + EDR

- FM : Bandes mondiales de 76 ~ 109 MHz avec pas de syntonisation 50KHz
- Dimensions: 24.0 x 24.0 x 2.6mm
- Poids: 3.30g
- Contrôle via les commandes AT (3GPP TS 27.007, 27.005 et commandes AT améliorées SIMCOM)
- Gamme de tension d'alimentation 3.4 ~ 4.4V
- Faible consommation d'énergie
- Température de fonctionnement : -40 ~ 85

#### II.3.3.4. Caractéristiques GPS

- 22 canaux de suivie
- Démarrage à froid : 32s
- Démarrage à chaud < 3s
- Démarrage très chaud < 1s
- Exactitude de positionnement horizontale inférieure à 2,5m CEP

#### II.3.3.5. Le montage de la carte WEMOS et le module SIM808

La figure II.8 représente le montage un système de localisation utilisant le module SIM808. Ce système est capable de déterminer la position du patient grâce à un récepteur GPS et permet d'envoyer ses informations de position par SMS vers le médecin, pour que cela fonctionne, il est nécessaire d'avoir une carte SIM (puce téléphonique).

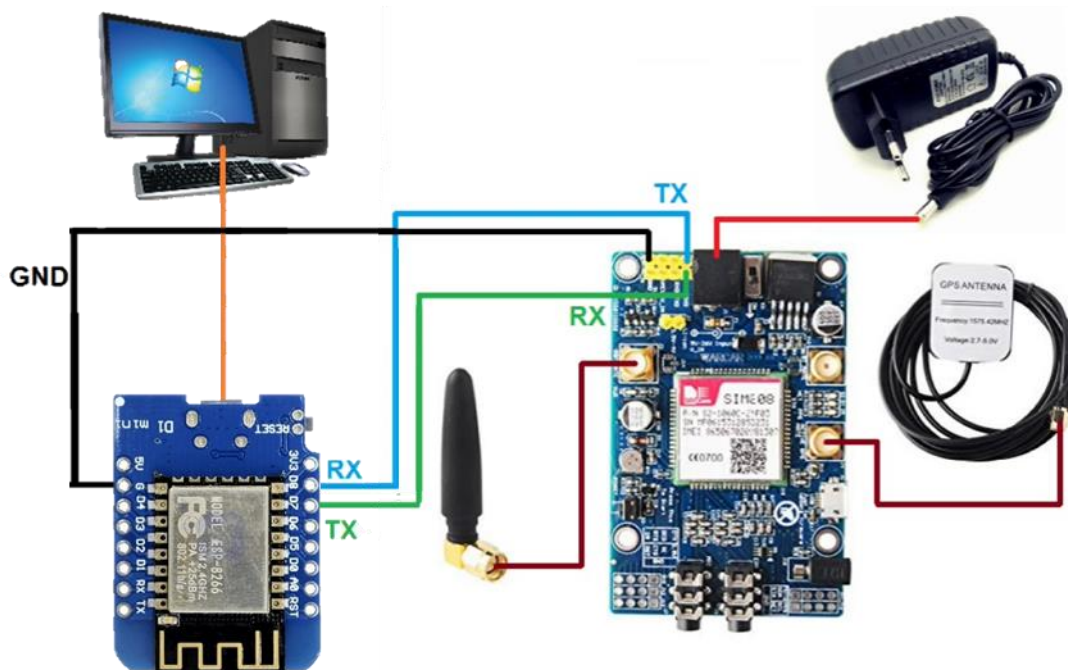


Figure II.8 : Communication du carte WEMOS et carte SIM808

### II.3.3.6. Les commandes AT

Les commandes AT contrôlent SIM808. Le tableau II.4 montre quelques commandes AT utilisées dans notre projet avec leur description.

commande AT	Description de la commande
AT	Match débit en bauds/Vérifier l'interface
ATI	Reconnaissance de bouclier
AT+CCID	Confirmation de la carte SIM
AT+CREG	Vérifier l'enregistrement du réseau
AT+CSQ	Fournir la qualité du signal
AT+CMGF	format de message texte
AT+CNMI	Pour spécifier comment traiter le nouveau message arrivé
AT+CMGS	Envoyer le message
AT+CMGR	Lire le message
AT+CGNSTST	Envoyer des données GNNS à AT UART
AT+CGNSPWR	Contrôle de puissance du GNSS
AT+CGNSSEQ	Définir les dernières phrases NMEA analysées
AT+CGNSINF	Lecture de phrases NMEA

*Tableau II.4 : Commande AT et leur description*

## II.3.4. Capteur de fréquence cardiaque ECG

### II.3.4.1. Description du signal électrocardiogramme (ECG)

Le signal électrocardiogramme (ECG) est une représentation graphique de l'activité électrique cardiaque qui représente les impulsions électriques. L'onde d'activation permet la repolarisation et la dépolarisation des cellules cardiaques qui peuvent être reçues par des électrodes (capteurs) fixées sur la peau à certains points dans le corps [23].

Cet enregistrement permet aux cardiologues de mesurer le rythme cardiaque et de détecter et diagnostiquer des anomalies cardiaques.

#### II.3.4.1.1. Composants du signal ECG

La morphologie ECG du tracé est simplement la forme d'onde ou la perspective de l'activité électrique du muscle cardiaque, de la dépolarisation et de la

repolarisation, dans un cycle cardiaque. La dépolarisation des cellules qui provoque la systole, la phase de contraction.

La Figure II.9 indique la forme d'onde globale dite ECG normale.

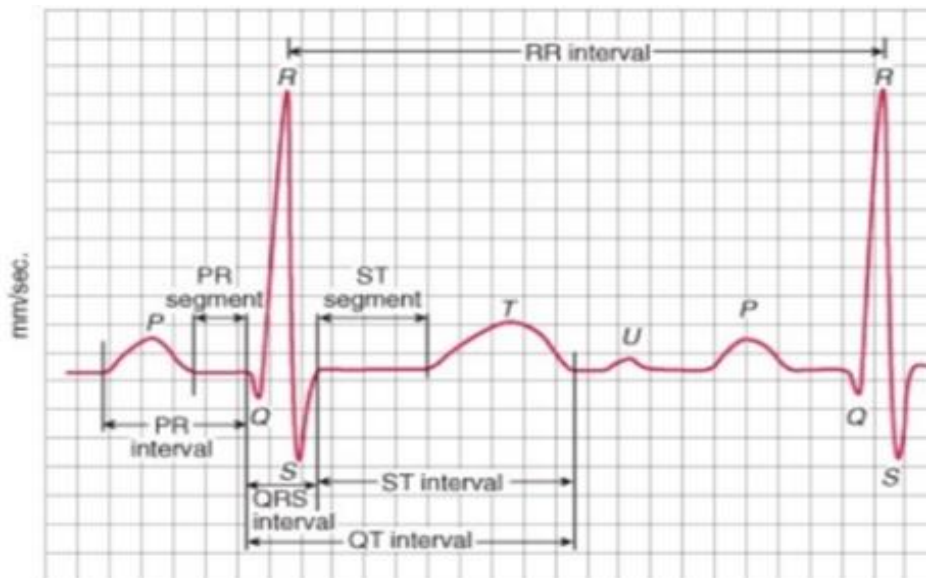


Figure II.9 : Le signal d'électrocardiogramme (ECG)

Les formes d'onde qui composent l'ECG sont décrites comme suit [24] :

- **L'onde P** se réfère à l'activation électrique de la dépolarisation auriculaire qui provoque la conduction de l'impulsion électrique à travers les oreillettes.
- **Le complexe QRS** montre une dépolarisation ventriculaire qui provoque une contraction des ventricules.
- **L'interval PR** commence à partir du début de l'onde P à un début du complexe QRS.
- **L'onde T** affiche la repolarisation des ventricules pendant le temps où les ventricules retournent à leur état électrique de repos.
- **L'interval QT** commence à partir d'un début du complexe QRS jusqu'à la fin de l'onde T. L'interval QT présente une dépolarisation et une repolarisation ventriculaires.
- **Le segment TP** commence à partir de la fin de l'onde T du cycle d'ECG (ou battement de cœur) précédent jusqu'au début de l'onde P du cycle ECG suivant. Le segment TP représente le moment où les cellules du muscle cardiaque sont électriquement silencieuses. Ainsi, il est toujours illustré par un intervalle isoelectrique qui représente une ligne zéro, une ligne de base ou une ligne électrique.
- **L'interval RR** il est délimité par deux pics R successives et d'où est évaluée la fréquence cardiaque instantanée. Cet intervalle est utilisé pour la détection des arythmies ainsi que pour l'étude de la variabilité de la fréquence cardiaque.

- **Le segment ST** il représente l'intervalle durant lequel les ventricules restent dans un état de dépolarisation actif. Il est aussi défini comme la durée entre la fin de l'onde S et le début de l'onde T.
- **L'onde S** elle représente la durée de dépolarisation ventriculaire (Les chambres en bas à droite et à gauche) précèdent l'effet mécanique de contraction. Sa durée normale est comprise entre 85ms et 95ms.
- **L'onde U** Signal électrique de base amplitude et de basse fréquence (« phénomène mécanoélectrique ») qui survient après l'onde T ou fusionne avec elle. Sa signification n'est pas bien connue, elle pourrait correspondre à l'onde T de repolarisation des cellules de Purkinje.

Les valeurs habituelles des différents paramètres de l'ECG :

	<b>Onde P</b>	<b>Complexe QRS</b>	<b>Onde T</b>	<b>Intervalle QT</b>	<b>Segment ST</b>	<b>Segment PR</b>
<b>L'amplitude (mv)</b>	< 0.25	0.5 - 2	0.18	0.1 - 0.5	Isoélectrique : 0	Isoélectrique : 0
<b>La durés (s)</b>	0.08-0.1	0.06 - 1	0.2	0.35-0.44	0.015 - 0.5	0.12 - 0.2

*Tableau II.5 : Les valeurs habituelles des différentes composantes du signal ECG [25]*

### II.3.4.1.2. Le rythme cardiaque

Le rythme cardiaque est caractérisé par deux propriétés : la fréquence des ondes R, exprimée en nombre de battements par minute (bpm), et leur régularité.

En l'absence de toute pathologie, le rythme est régulier et sa fréquence est comprise entre 60 et 100 bpm. En cas de dépassement de ces limites, il peut y avoir un trouble du rythme (les arythmies cardiaques) qui doit faire l'objet d'une étude approfondie pour définir une éventuelle pathologie [26].

Les arythmies cardiaques sont une cause majeure de mortalité et constituent une partie très importante de la problématique des maladies cardiovasculaires.

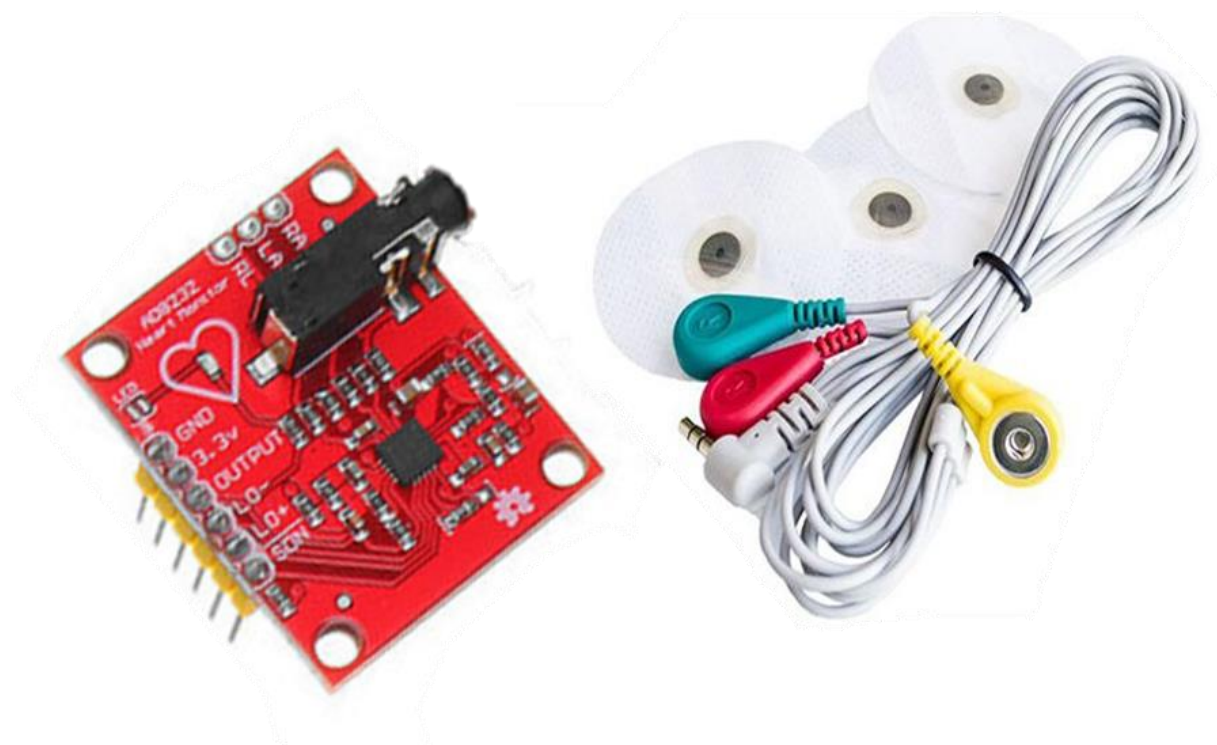
On classe les arythmies en fonction de leur lieu de formation et de leurs effets sur le rythme cardiaque.

- Si une arythmie accélère le rythme cardiaque, on parlera de tachycardie (tachy = rapide).
- Si une arythmie ralentit, on parlera alors de bradycardie (brady = lent).

### II.3.4.2. Le module AD8232 (capteur ECG)

Le module AD8232 (moniteur de fréquence cardiaque) est une petite puce utilisée pour mesurer l'activité électrique du cœur. Cette activité électrique peut être représentée sous forme d'électrocardiogramme (ECG), et la sortie de ce module est lue sous forme analogique. L'électrocardiographie est utilisée pour aider à diagnostiquer diverses maladies cardiaques [27].

Module AD8232 en vente contient la carte AD8232 et 3 électrodes à usage unique (disponible sur marché) et un câble conducteur adapté.



*Figure II.10 : Capteur de fréquence cardiaque ECG (module AD8232)*

La mesure se fait à l'aide de 3 électrodes placées sur la peau aux endroits suivants :

1. Bras droit sur le devant de l'avant-bras ou sur le côté droit de la poitrine.
2. Bras gauche sur le devant de l'avant-bras ou sur le côté gauche de la poitrine.
3. Jambe droite au milieu de la cuisse ou au bas de l'abdomen droit.



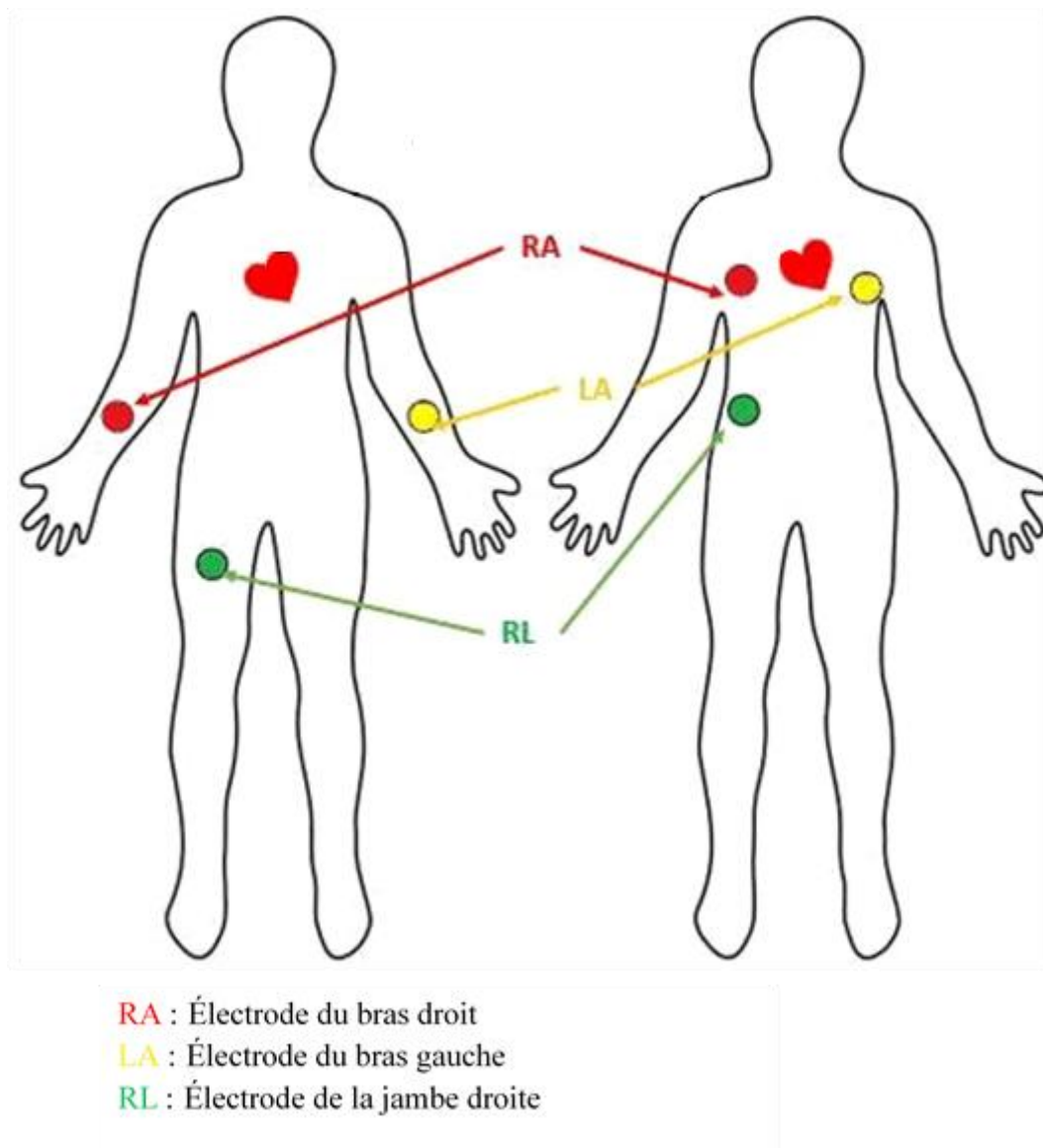


Figure II.11 : L'emplacement des électrodes dans le corps

### II.3.4.3. Caractéristiques de capteur ECG

Parmi les caractéristiques les plus importantes de module AD8232 [28] :

- AD8232 adopte un filtre passe-haut à deux pôles pour éliminer les artefacts de mouvement et le potentiel de demi-cellule d'électrode.
- AD8232 adopte un amplificateur opérationnel sans contrainte pour construire un filtre passe-bas à trois pôles, éliminant les bruits supplémentaires.
- Plage de température nominale :  $[0^{\circ}; 70^{\circ}]$ .
- Plage de température de travail :  $[-40^{\circ}; 85^{\circ}]$ .
- SDN, LO +, LO-, SORTIE, 3.3V, GND fournissent les broches essentielles au fonctionnement de ce moniteur avec une carte Arduino ou une autre carte de développement.
- Cette carte contient également des broches RA (bras droit), LA (bras gauche) et RL (jambe droite) pour fixer et utiliser nos propres capteurs personnalisés.

▪ De plus, il y a un voyant lumineux à LED qui pulsera au rythme d'un battement de cœur.

#### II.3.4.4. Connexion du capteur ECG et la carte WEMOS D1 Mini

Le tableau II.6 et la figure II.12 montrent la connexion du module AD8232 et le module WEMOS D1 Mini.

Broche	Fonction de broche	Broche WEMOS
GND	Masse (Ground)	GND
3.3V	Alimentation 3.3V	3.3V
Output	Signal de sortie	A0
LO+	Détection de dérivation + (Lead-Off+)	D5
LO-	Détection de dérivation - (Lead-Off -)	D6
LOSDN	Fermer (Shutdown)	Non utilisé

Tableau II.6 : Connexion du capteur ECG et la carte WEMOS D1 Mini [29]

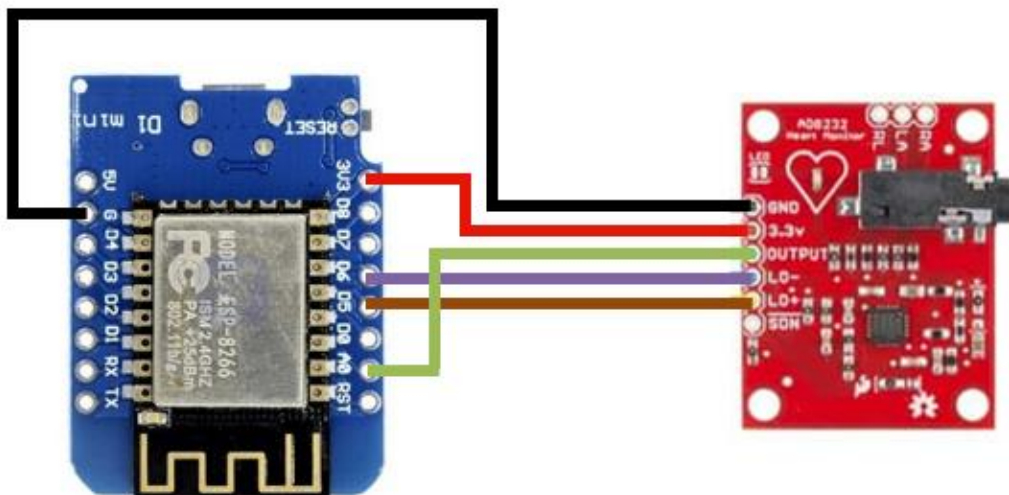


Figure II.12 : Connexion du capteur ECG et la carte WEMOS D1 Mini

#### II.3.5. LEDs et résistances protectrices

Une diode électroluminescente plus connue sous l'appellation Del ou Led désigne un dispositif ou composant optoélectronique qui émet de la lumière lorsqu'un courant électrique la traverse.

La résistance est un dipôle qui s'oppose au passage du courant électrique qui la traverse, plus une résistance possède une valeur élevée et plus celle-ci résiste à la circulation du courant.



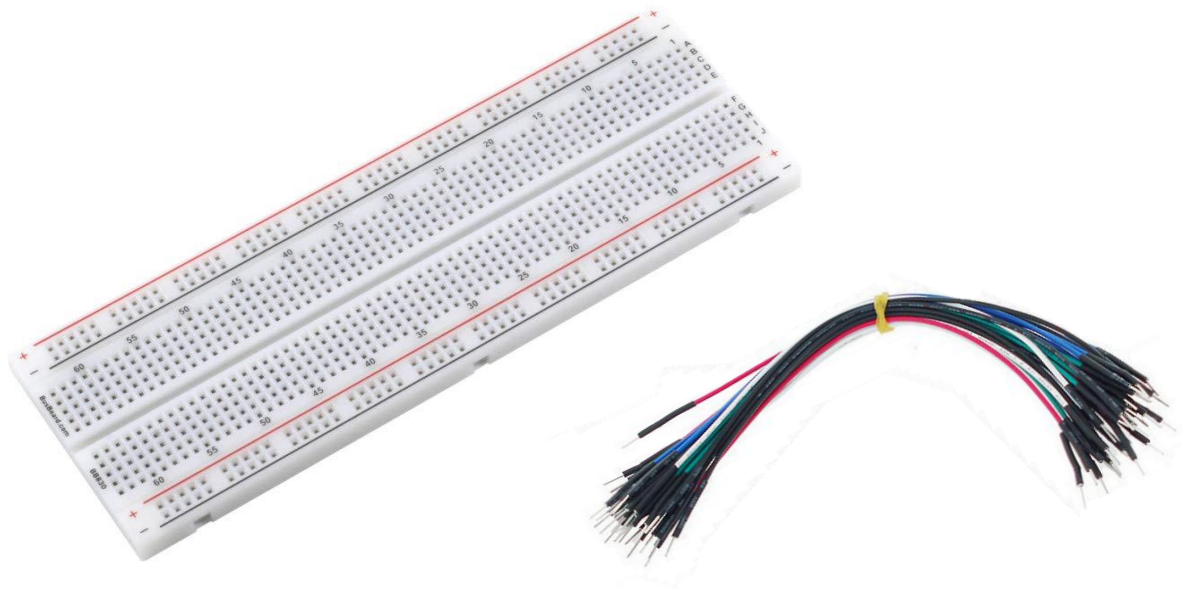
*Figure II.13 : Les LEDs et Les Résistances protectrices*

Les résistances sont notamment utilisées pour protéger les diodes électroluminescentes (DEL) qui ne supportent qu'une tension faible.

### **II.3.6. Plaque d'essai et câbles de démarrage (Jumper cables)**

La plaque d'essai sans soudures (BreadBoard) est une plaquette utilisée comme un outil pédagogique indispensable pour découvrir l'électronique. Son principal avantage est de permettre de tester et réaliser des montages rapidement des circuits électroniques sans souder aucuns composants. Il est donc possible de réutiliser les composants.

Les câbles jumper servent également à relier les composants électroniques les uns aux autres sur la plaque d'essai.



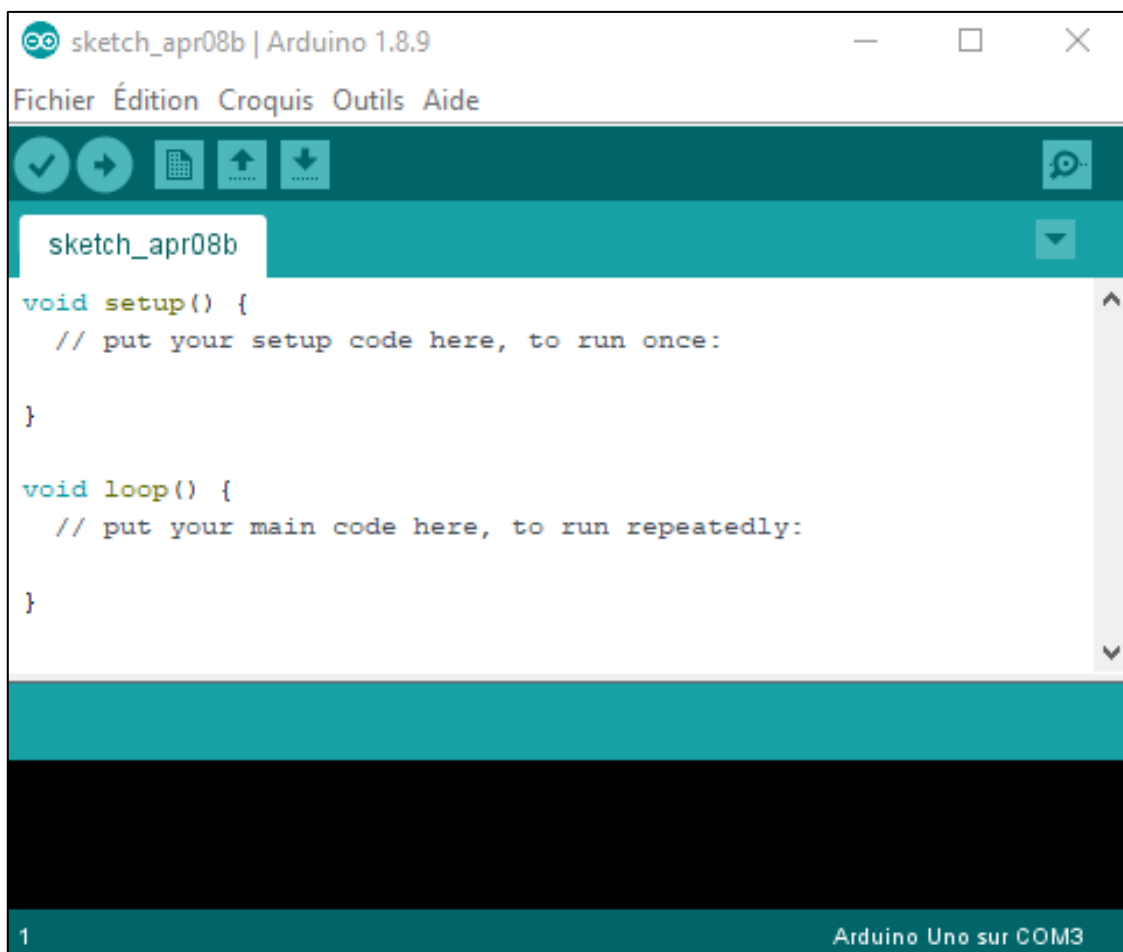
*Figure II.14 : Plaque d'essai sans soudure et câbles Jumper*

## II.4. Logiciels et plateformes requis

### II.4.1. L'environnement de développement Arduino

#### II.4.1.1. Description du logiciel Arduino IDE

Le logiciel Arduino IDE (Integrated Development Environment) est une application Java, libre et multi-plateforme qui permet de programmer les différents modules Arduino et utiliser leurs entrées/sorties, avec un langage propre à lui dont la structure s'apparente aux langages C/C++.



*Figure II.15 : La fenêtre principale du logiciel Arduino IDE.*

Le logiciel Arduino a pour fonctions principales :

- De pouvoir écrire et compiler des programmes pour la carte Arduino.
- De se connecter avec la carte Arduino pour transférer les programmes.
- De communiquer avec la carte Arduino.

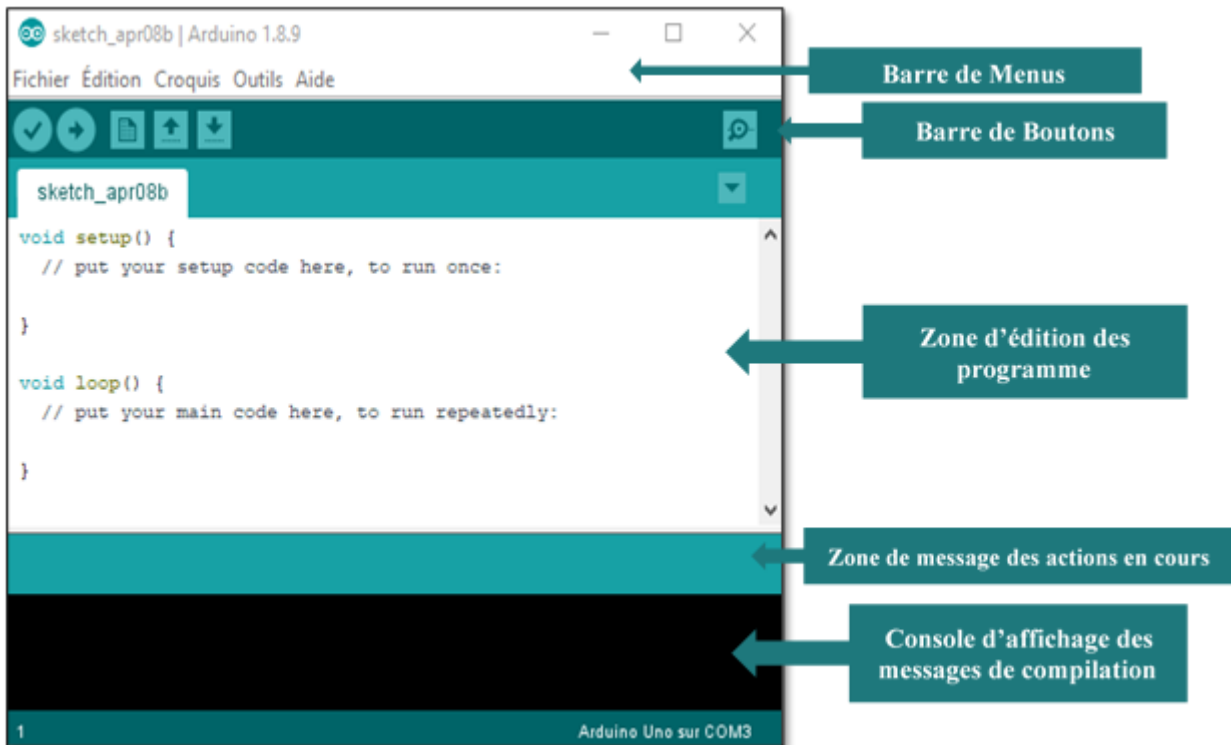


Figure II.16 : Interface Arduino IDE.

Ce logiciel comporte :

- **Une barre de menus** : comme pour tout logiciel une interface graphique (GUI)
- **Une barre de Boutons** : qui donne un accès direct aux fonctions essentielles du logiciel (compiler, téléverser ...).
- **Un Editeur** : pour écrire le code de notre programme.
- **Une Zone de Message** : qui affiche et indique l'état des actions en cours.
- **Une Console Texte** : qui affiche les messages de résultat de la compilation du programme.
- **Un Moniteur série (terminal)** : qui permet d'afficher des messages textes reçus de la carte Arduino et d'envoyer des caractères vers la carte Arduino. Cette fonctionnalité permet d'afficher sur l'ordinateur l'état de variables, de résultats ou de calculs. Le moniteur est un élément essentiel pour améliorer, tester et corriger les programmes.

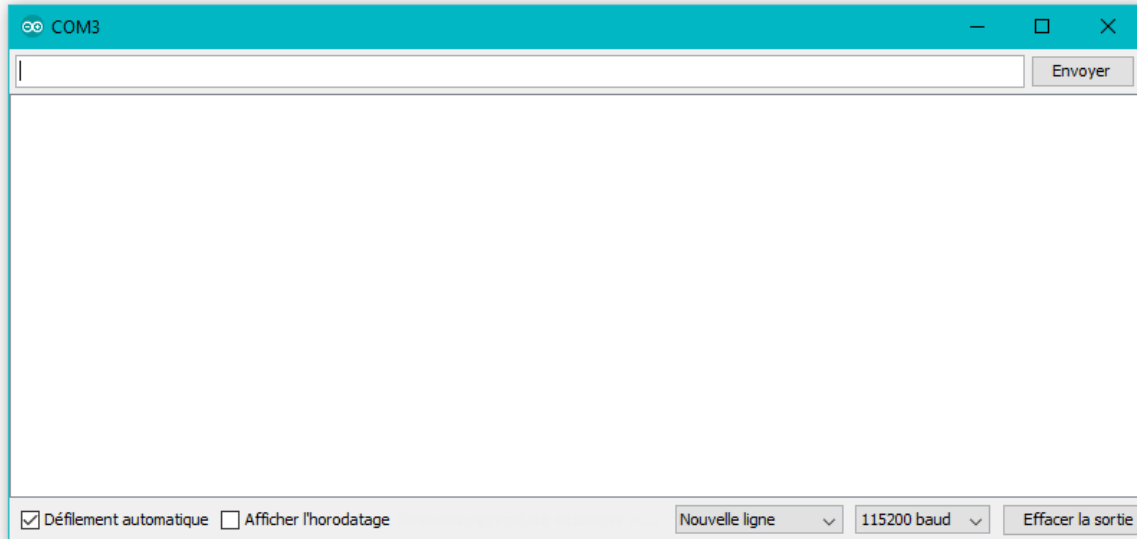


Figure II.17 : Fenêtre de moniteur série

▪ **Un Traceur série** : affiche des valeurs reçues de la carte Arduino sous forme d'un graphe pour faciliter l'observation de nos résultats.

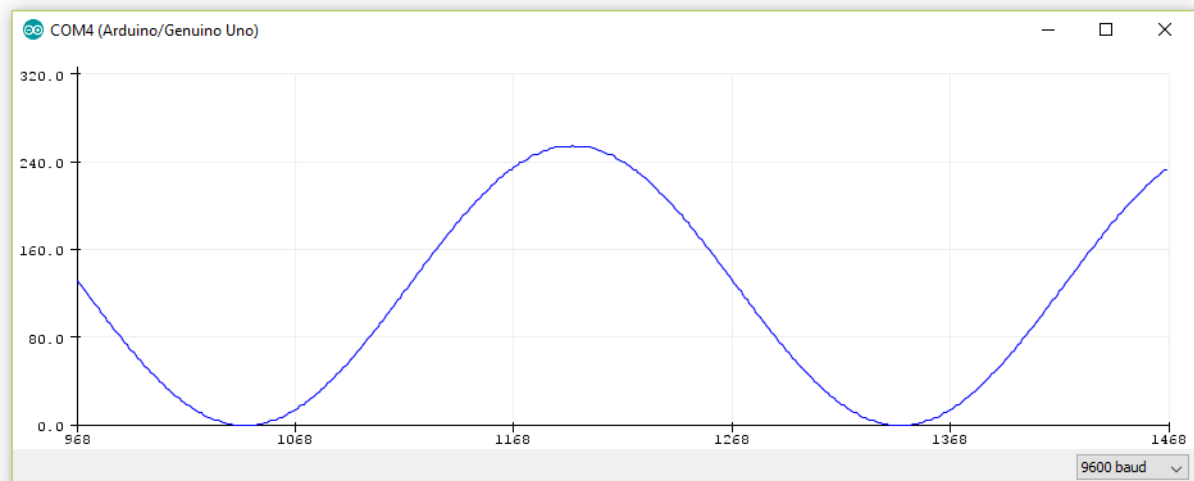


Figure II.18 : Fenêtre de Traceur série

## II.4.1.2. Syntaxe du langage Arduino

### II.4.1.2.1. Commandes de structure du programme

- Structure générale :

```
void setup() {  
  (configuration-préparation)  
}  
void loop() {  
  (exécution)  
}
```

- Variables :

char (variable ‘caractère’)

int (variable ‘nombre entier’)

long (variable ‘nombre entier de très grande taille’)

string (variable ‘chaîne de caractères’)

array (tableau de variables)

- Niveaux logiques des connecteurs numériques :

HIGH (état 1)

LOW (état 0)

INPUT (configuré en entrée)

OUTPUT (configuré en sortie)

- Autres commandes :

// (commentaire simple ligne)

/\* \*/ (commentaire multi-lignes)

#define ( donner une valeur à un nom)

#### **II.4.1.2.2. Opérations de comparaison**

- Opérations de Contrôle :

if (si...)

if...else (si...alors...)

for (pour...)

switch case (dans le cas où...)

while (pendant que...)

- Les conditions :

!= (différent de)

< (inférieur à)

> (supérieur à)

<= (inférieur ou égal à)

>= (supérieur ou égal à)

- Opérations booléennes :

&& (et)

|| (ou)

! (non)

---

### II.4.1.2.3. Fonctions

- Entrées-sorties numériques :

pinMode(broche, état) → (configuration des broches)

digitalWrite(broche, état) → (écrire un état sur une broche num.)

digitalRead(broche) → (lire un état sur une broche num.)

unsigned long pulseIn(broche, état) → (lire une impulsion sur une broche num.)

- Entrées analogiques :

int analogRead(broche) → (lire la valeur d'une broche ana.)

analogWrite(broche, valeur) → (PWM : écrire une valeur analogique sur les broches 9, 10 ou 11)

- Gestion du temps :

unsigned long millis() → (temps de fonctionnement du programme)

delay(ms) → (attente, en millisecondes)

- Nombres aléatoires :

randomSeed(seed) → (aléatoire 'pilote')

long random(max) → (aléatoire à partir de telle valeur)

long random(min, max) → (aléatoire entre deux valeurs)

- Communications série entre Arduino et autres module

Serial.begin(speed) → (configuration de la vitesse de communication Série)

Serial.available() → (donne combien de caractères disponibles dans la zone tampon Série)

Serial.read() → (lit les données Série)

Serial.print(data) → (envoi des données Série)

Serial.println(data) → (envoi des données Série suivies de caractères spécifiques.)

## II.4.2. Application web

Nous avons choisi le logiciel Visual Studio comme environnement de développement. Visual Studio offre une prise en charge intégrée de JavaScript. Il offre la complétion de code ainsi qu'une fonction de refactorisation du code. Son débogueur peut fonctionner aussi bien au niveau de la machine qu'au niveau du code source.

Le concepteur de classes, le profileur de code, le concepteur de schémas de base de données, le concepteur de formulaires et le concepteur web sont



disponibles sous forme d'outils intégrés à Visual Studio. Pour étendre encore ses capacités, un certain nombre de plug-ins sont disponibles.

Nous avons développé notre application web avec les langages :

#### **II.4.2.1. HTML**

HTML est un langage informatique utilisé sur Internet. Ce langage est utilisé pour créer des pages Web. L'acronyme signifie HyperText Markup Language, qui signifie en français "langage de balisage d'hypertexte". Cette signification porte bien son nom puisqu'effectivement ce langage permet de réaliser de l'hypertexte à base d'une structure de balisage.

#### **II.4.2.2. CSS**

Le CSS pour Cascading Style Sheets, est un langage informatique utilisé sur Internet pour la mise en forme de fichiers et de pages HTML. On le traduit en français par feuilles de style en cascade.

#### **II.4.2.3. JavaScript**

Le JavaScript est un langage de développement informatique, et plus précisément un langage de script orienté objet. On le retrouve principalement dans les pages Internet. Il permet, entre autres, d'introduire sur une page web ou HTML des petites animations ou des effets.

JavaScript est un langage de script qui offre de nombreuses possibilités. Sa présence croissante sur le web permet de naviguer sur des sites de plus en plus interactifs et rapides.

### **II.4.3. Cloud Computing**

#### **II.4.3.1. Définition**

Le cloud computing ou informatique en nuage est une infrastructure dans laquelle la puissance de calcul et le stockage sont gérés par des serveurs distants auxquels les utilisateurs se connectent via une connexion Internet sécurisée.

Selon la National Institute of Standards and Technology, le Cloud Computing est un modèle qui permet l'accès à un réseau à la demande à un ensemble de ressources informatiques mutualisées et configurables (ex réseaux, serveurs, stockage, applications et services). Ces ressources informatiques peuvent être allouées et libérées rapidement avec le minimum d'effort de gestion ou d'interaction avec les fournisseurs de services [30].

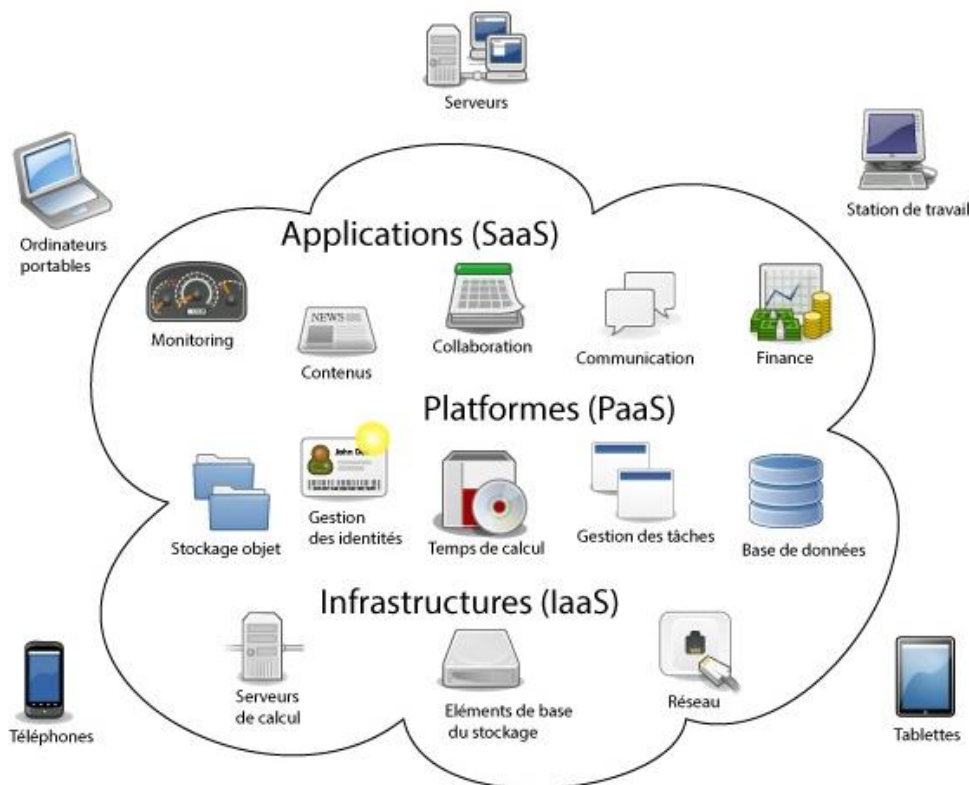


Figure II.19 : Cloud Computing

### II.4.3.2. Caractéristiques

Le Cloud Computing est composé de cinq caractéristiques essentielles [31] :

- **Ressources à la demande :** Un utilisateur peut allouer unilatéralement des ressources informatiques (serveurs, réseau, stockage, environnement d'exécution, application) au besoin, de façon automatique et sans nécessité d'interaction humaine avec chaque fournisseur de services.
- **Large accès réseau :** Les ressources Cloud Computing sont disponibles à travers le réseau et accessibles via des mécanismes standards qui favorisent leurs utilisations à partir des appareils clients hétérogènes, voire légères (ex ordinateurs portables, téléphones, tablettes).
- **Mutualisation des ressources :** Les ressources informatiques du fournisseur Cloud Computing sont mutualisées pour servir plusieurs clients en utilisant un modèle multi-tenant. Ces ressources, physiques ou virtuelles, sont allouées et libérées dynamiquement selon la demande du consommateur. Généralement, l'utilisateur n'a ni le contrôle ni la connaissance de l'emplacement exact des ressources allouées. Dans certains cas, il peut choisir l'emplacement géographique à un niveau haut (ex par pays, continent ou data-center).
- **Élasticité rapide :** Les ressources sont allouées et libérées d'une façon élastique, idéalement d'une façon automatiquement, pour s'adapter rapidement à la demande qu'elle soit croissante ou décroissante. Pour le consommateur, les

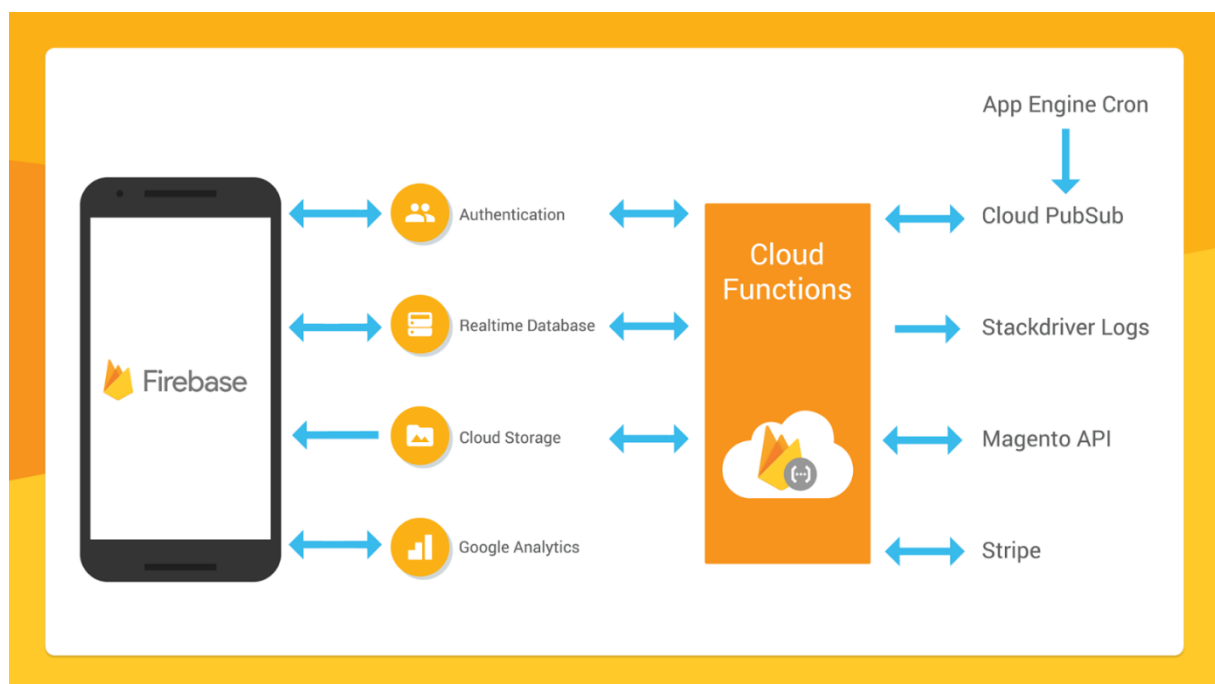
ressources disponibles à l'allocation apparaissent comme illimitées et peuvent s'allouer à tout moment.

- **Services mesurés :** Toutes les ressources allouées peuvent être surveillées et contrôlées afin de mesurer leurs consommations avec un niveau d'abstraction approprié selon le type du service (ex stockage, temps de calcul, bande passante).

### II.4.3.3. Firebase

Firebase est un ensemble de services d'hébergement pour n'importe quel type d'application (Android, iOS, Javascript, Node.js, Java, Unity, PHP, C++...). Il propose d'héberger en NoSQL et en temps réel des bases de données, du contenu, de l'authentification sociale (Google, Facebook, Twitter et Github), et des notifications, ou encore des services, tel que par exemple un serveur de communication temps réel.

Firebase Realtime Database n'est autre qu'une base de données NoSQL, bénéficiant d'un hébergement «Cloud» et permettant le stockage et la synchronisation de données des utilisateurs. Les développeurs peuvent gérer cette base de données en temps réel.



*Figure II.20 : Firebase cloud*

## **II.5. Conclusion**

Dans ce chapitre, nous avons décrit les caractéristiques techniques et fonctionnelles de tous les matériels et logiciels utilisés dans le développement de notre système de surveillance, avec une définition générale sur les signaux cardiaques (ECG).

Dans le chapitre suivant, nous présenterons en détail la réalisation pratique de notre projet IoT dédié pour le suivi des patients cardiaques.

# *Chapitre III :*

*La réalisation pratique du système IoT*

### III.1. Introduction

A travers ce chapitre, nous allons expliquer en détail la conception de notre projet basé sur le suivi des patients cardiaques, en commençant par le montage du système, puis, les algorithmes de fonctionnement du système, et à la fin du chapitre nous allons montrer les résultats obtenus.

### III.2. Montage du système

Comme nous l'avons mentionné au chapitre 2, le système de surveillance que nous avons réalisé comporte les éléments suivants :

- Carte WEMOS D1 Mini (ESP8266)
- Module GSM/GPRS/GPS (SIM808)
- Capteur de fréquence cardiaque (AD8232).

Ces éléments sont connectés comme indiqué dans la figure III.1 et le tableau III.1.

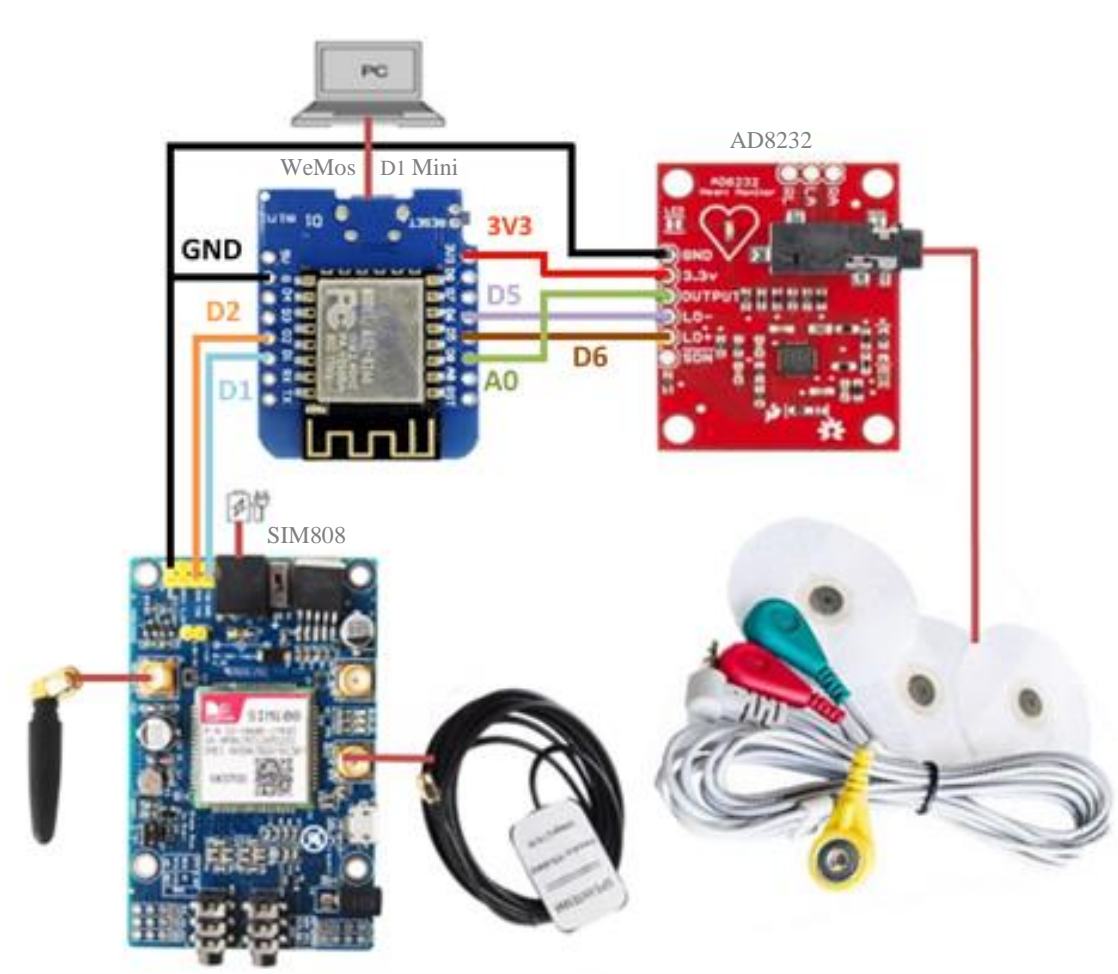


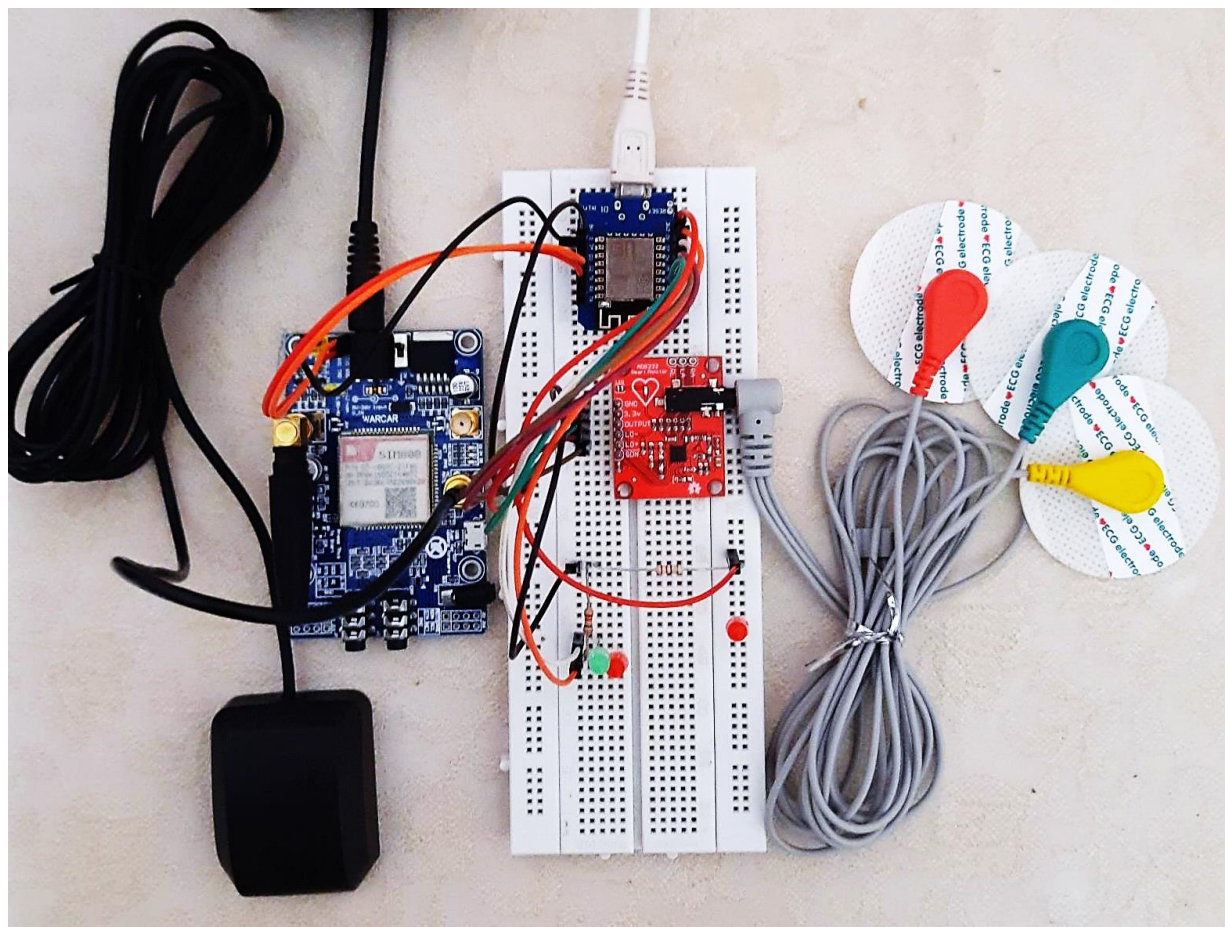
Figure III.1 : Circuit du Système

	Broche	Broche WEMOS
<b>AD8232</b>	GND	GND
	3.3V	3.3V
	Output	A0
	LO+	D5
	LO-	D6
<b>SIM808</b>	GND	GND
	TX	D2
	RX	D1

*Tableau III.1 : Connection de différentes broches de la carte WEMOS*

Nous avons ajouté des LED uniquement pour montrer le fonctionnement de certains paramètres.

La figure III.2 représente le montage de notre système en pratique.



*Figure III.2 : Le montage de notre système*

### III.3. Installation de la carte ESP8266 dans l'Arduino IDE

La communauté ESP8266 a créé un module complémentaire pour l'Arduino IDE qui nous permet de programmer l'ESP8266 à l'aide de l'Arduino IDE.

Avant de commencer cette procédure d'installation, nous devons installer la dernière version de l'Arduino IDE à partir de "arduino.cc/fr/Main/Softwar".

Pour installer la carte ESP8266 dans notre Arduino IDE, on suit les instructions suivantes :

✓ Dans l'Arduino IDE, allez dans Fichier > Préférences, mettez le lien 'http://arduino.esp8266.com/stable/package\_esp8266com\_index.json' dans le champ «URL de gestionnaires de cartes supplémentaires » comme indiqué dans la figure III.3. Ensuite, Cliquez sur le bouton « OK ».

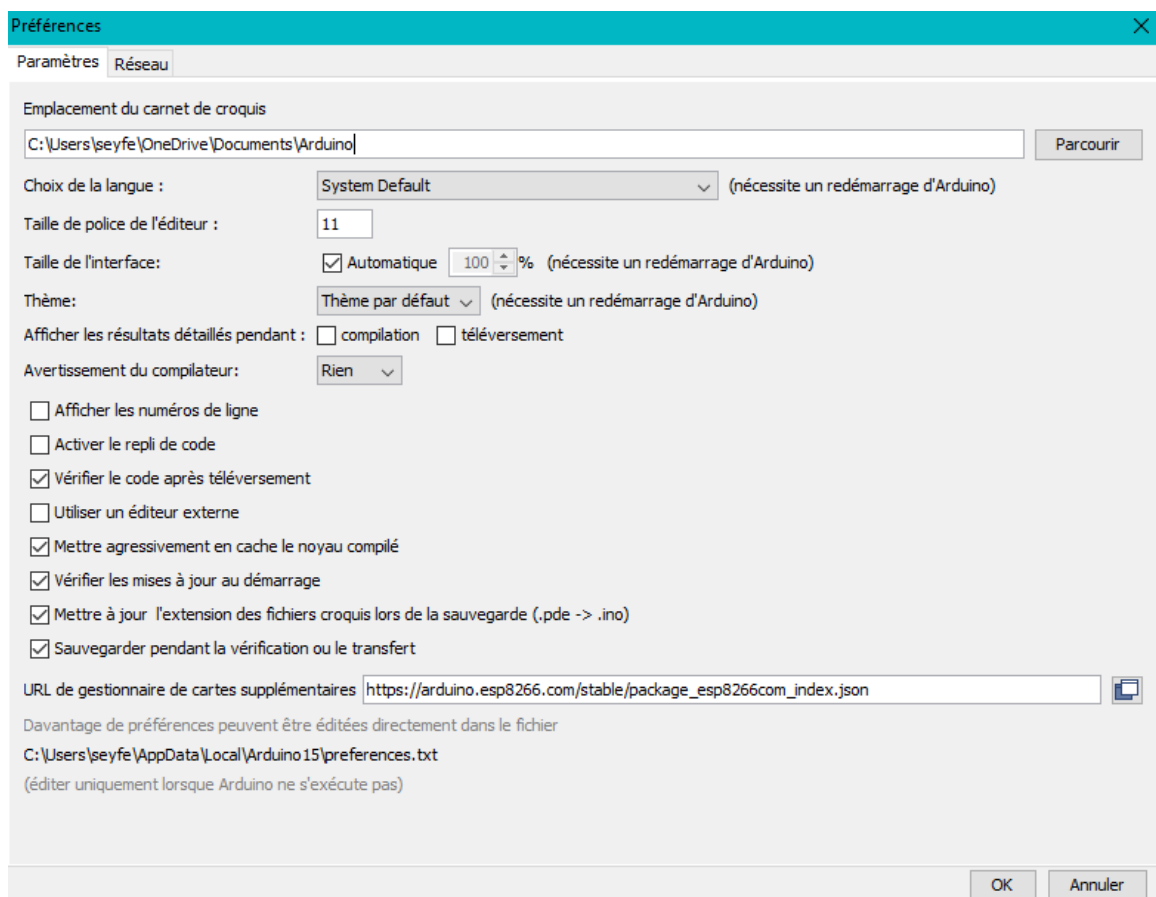


Figure III.3 : Fenêtre de Préférences de l'Arduino IDE

✓ Ouvrir le gestionnaire de cartes, cliquez à Outils > Gestionnaire de cartes. Recherchez ESP8266 et cliquez sur le bouton d'installation pour "ESP8266 by ESP8266 Community".



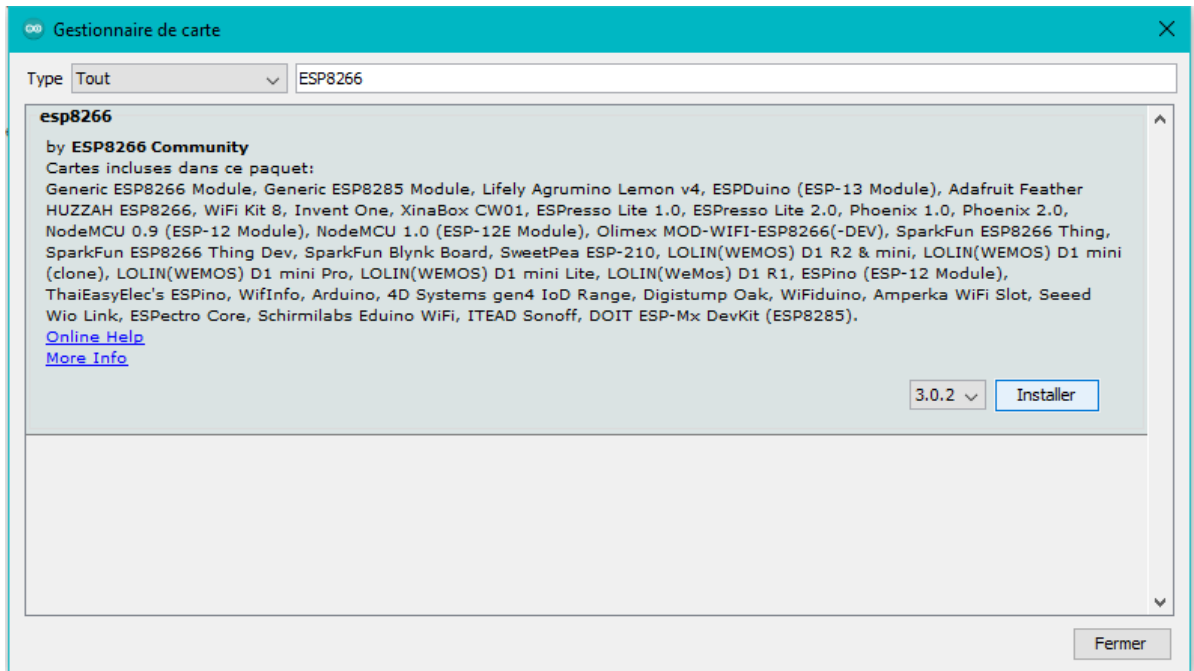


Figure III.4 : Fenêtre de gestionnaire de cartes de l'Arduino IDE

Le module complémentaire de carte ESP8266 devrait être installé dans l'Arduino IDE après quelques secondes.

Pour sélectionner la carte « WEMOS D1 mini », il suffit de cliquer sur Outils>Type de carte> ESP8266 Boards (3.0.2), puis sélectionner LOLIN(WEMOS) D1 R2 & mini.

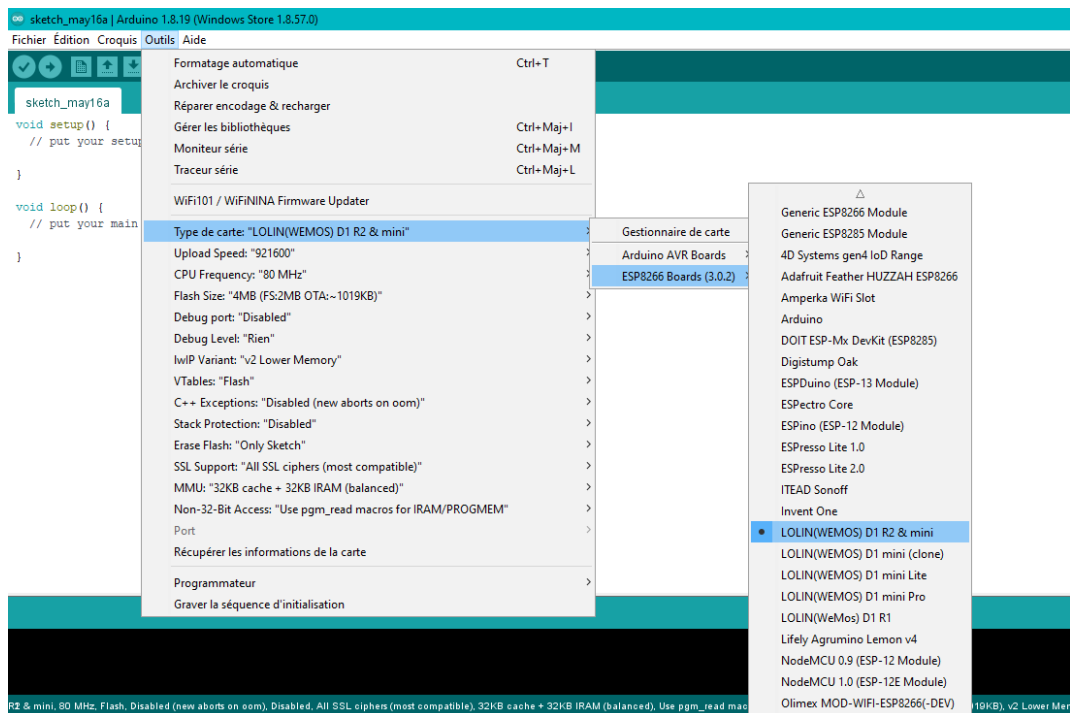


Figure III.5 : Sélection de la carte WEMOS D1 mini.

---

### III.4. L'algorithme global du fonctionnement du système

Le principe de fonctionnement de ce système est de lire le signal ECG capté par le capteur AD8232 et de compter le nombre de battements par minute et le comparer aux valeurs normales pour détecter et diagnostiquer les anomalies cardiaques puis de l'envoyer vers le cloud via le module WEMOS qui nous fournit la connexion WiFi. D'autre part, le système est doté d'un module GSM/GPRS/GPS qui assure la géolocalisation par GPS et permet d'envoyer des alertes SMS au médecin en cas d'anomalies observées chez le patient.

L'algorithme global du fonctionnement du système contient deux fonctions principales, ainsi que l'importation de bibliothèques.

#### III.4.1. Importation de bibliothèques

Nous avons importé les bibliothèques suivantes :

- La bibliothèque "<EEPROM.h>" permet d'accéder à la mémoire WEMOS pour stocker des valeurs.
- La bibliothèque "<ESP8266WiFi.h>" permet de connecter le WEMOS au réseaux WiFi.
- La bibliothèque "<ESP8266WebServer.h>" permet au WEMOS de se connecter en tant que serveur Web, cela prend en charge les requêtes HTTP GET et POST et exécute un client à la fois.
- La bibliothèque "<SoftwareSerial.h>" permet la communication série sur d'autres broches numériques de la carte WEMOS.
- La bibliothèque "<FirebaseESP8266.h>" fait la configuration firebase, les bibliothèques "<addons/TokenHelper.h>" et "<addons/RTDBHelper.h>" sont des bibliothèques supplémentaires à la bibliothèque firebase.

#### III.4.2. La fonction setup

Dans cette fonction, nous avons configuré la connexion série à la vitesse 19200 bauds pour l'ESP8266 et le SIM808, configuré les broches d'E/S, défini les paramètres WiFi et configuré la base de données Firebase. La fonction setup n'est exécutée qu'une seule fois, après chaque mise sous tension ou réinitialisation de la carte WEMOS.

#### III.4.3. La fonction loop

Cette fonction s'exécute dans une boucle sans fin, dans laquelle il est vérifié que les électrodes sont d'abord placées sur le corps, afin que la fonction de lecture de l'électrocardiogramme soit appelée pour calculer le nombre de battements cardiaques pour le comparer aux valeurs normales et déterminer l'état du patient, puis toutes les valeurs sont envoyées dans le cloud Firebase.

III.4.4. L'organigramme global

La figure III.6 représente l'organigramme global de notre système.

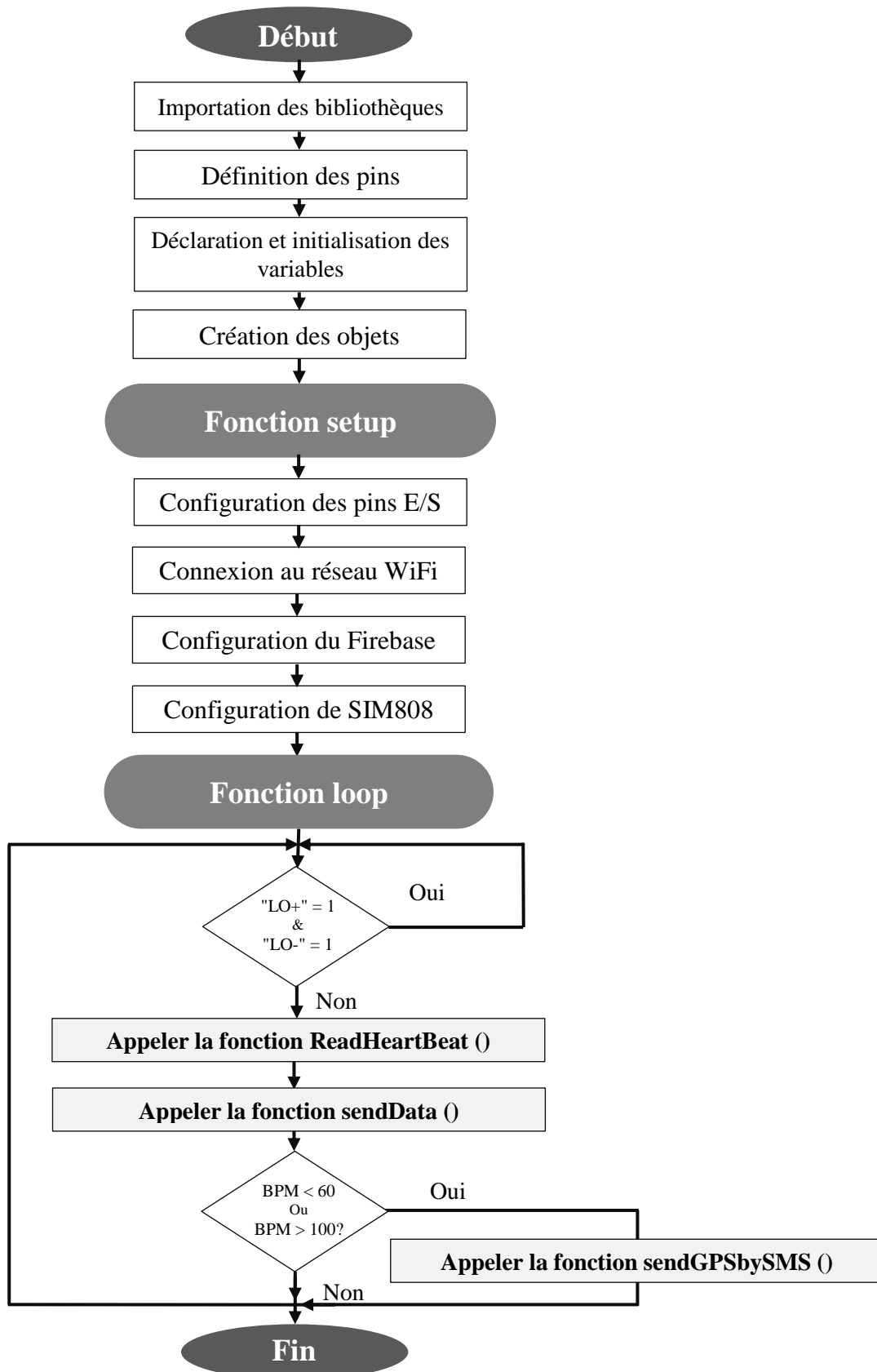


Figure III.6 : Organigramme global du système

### III.5. L'algorithme d'acquisition du signal ECG

La fréquence cardiaque est calculée à l'aide de la fonction ReadHeartBeat() en comptant le nombre de battements pendant dix secondes et en le multipliant par 6. L'algorithme de cette fonction est illustré par l'organigramme de la figure III.7.

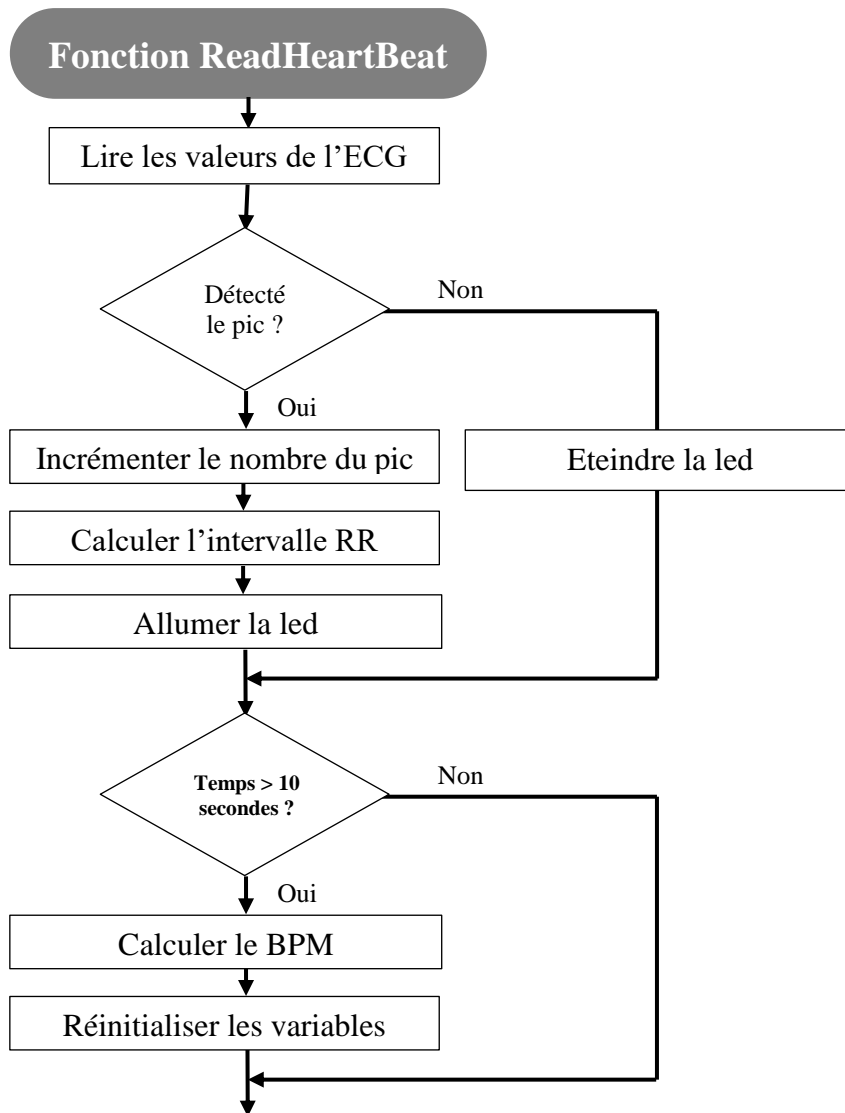


Figure III.7 : Organigramme de la fonction de calcul de battement

### III.6. L'algorithme d'envoyer les valeurs à Firebase Cloud

Le principe de la fonction sendData() est de stocker les valeurs d'ECG dans des chaînes de caractères et de les envoyer à firebase toutes les 10 secondes. Le fonctionnement de cet algorithme est décrit par l'organigramme de la figure III.8.

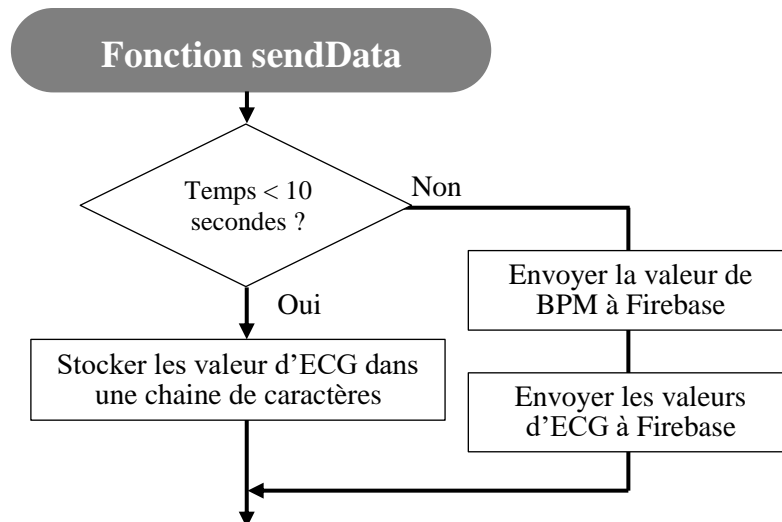


Figure III.8 : Organigramme de la fonction qui envoie les valeurs à Firebase Cloud

### III.7. L'algorithme d'acquisition du GPS et d'envoyer les SMS

Le rôle principale de la fonction sendGPSbySMS() est de lire la valeur de GPS puis de l'envoyer par SMS. Les informations de navigation GPS sont analysées à partir de phrase NMEA reçues par le SIM808 après l'exécution de la commande "AT + CGNSINF". La phrase NMEA incluent de nombreux paramètres, notamment : <Statut d'exécution GNSS>, <Statut fixe>, <Date et heure UTC>, <Latitude>, <Longitude>, etc., ces paramètres nous indiquent la date et la position GPS. Comme indiqué à l'organigramme de la figure III.7.

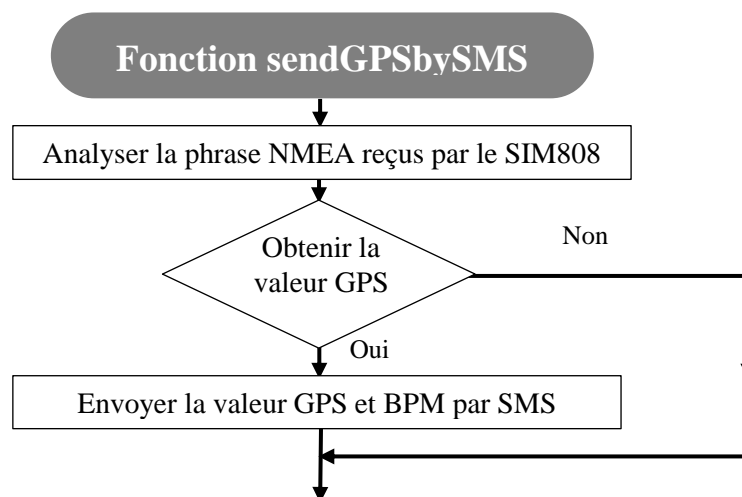


Figure III.9 : Organigramme de la fonction d'acquisition du GPS et d'envoi des SMS

### III.8. Création d'un projet Firebase

Pour créer un projet Firebase, nous avons suivi les étapes suivantes :

- ✓ Tout d'abord, nous devons avoir un compte Gmail.
- ✓ Après nous être connectés à la console Google Firebase, nous avons cliqué sur Ajouter un projet.

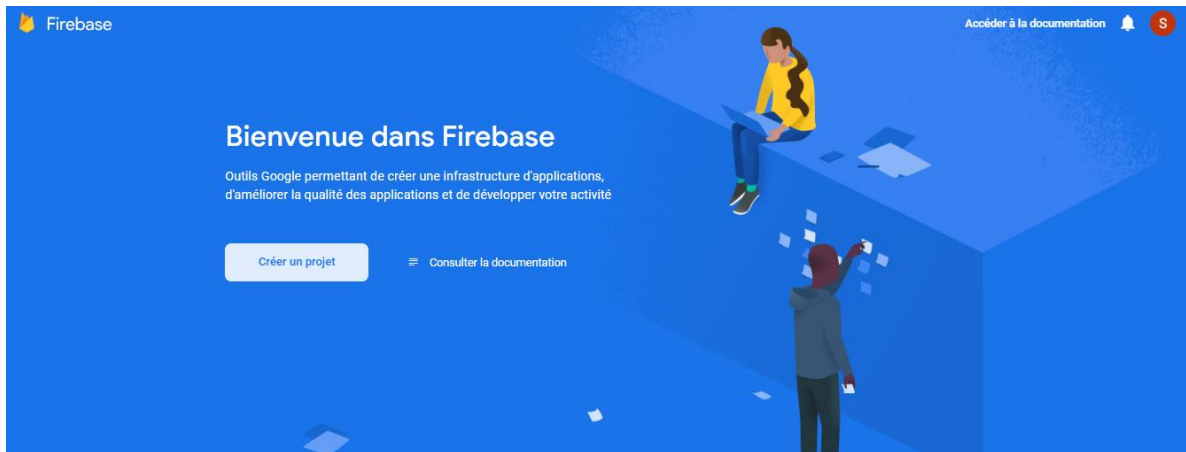


Figure III.10 : Création d'un projet Firebase

- ✓ Nous avons nommé notre projet et cliqué sur "Continuer".

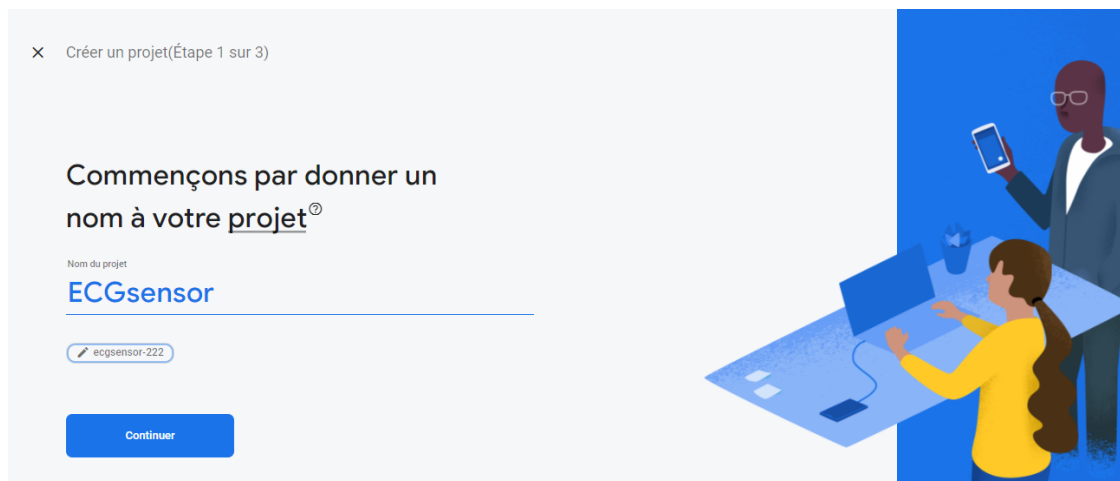


Figure III.11 : Créer un projet (Étape 1)

- ✓ Nous avons désactivé le compte Google Analytics pour ce projet, puis cliqué sur "Créer un projet".

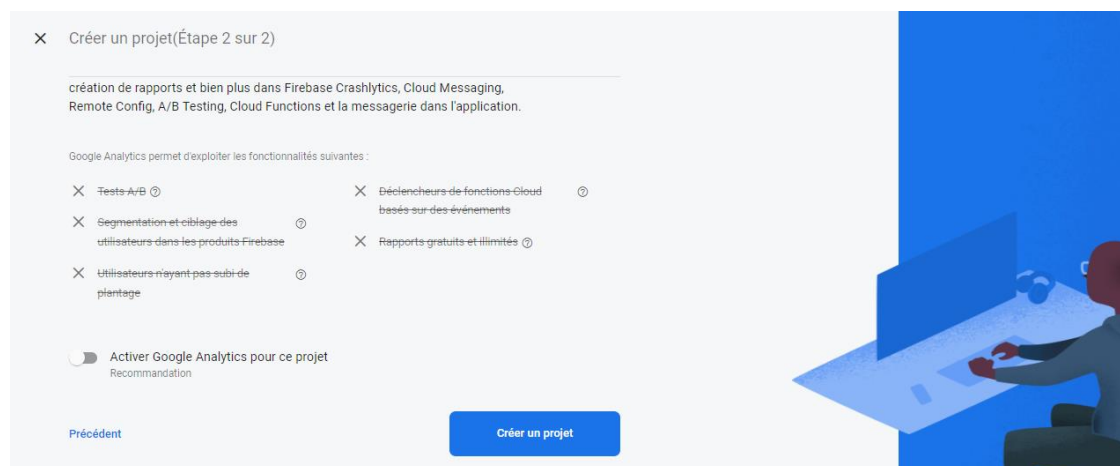
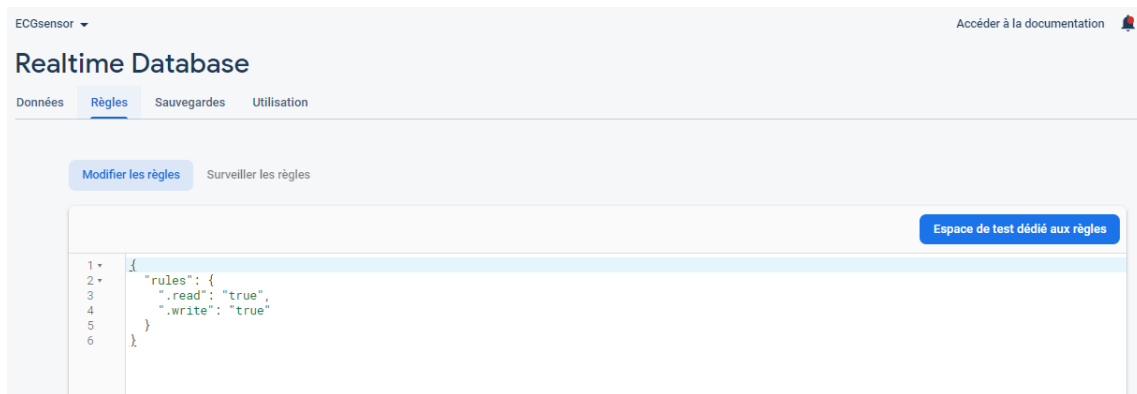


Figure III.12 : Créer un projet (Étape 2)



Figure III.14 : Modifier les règles pour Realtime Database



Le principe de fonctionnement de cette application est d'importer les données de Firebase en temps réel et de les lire, puis, afficher la fréquence cardiaque et le signal ECG sous forme graphique à l'aide de la bibliothèque Plotly.

### III.10. Résultats obtenus

Lors de la première utilisation du système, le composant WEMOS crée son propre réseau WiFi, et il se connecte à ce réseau en tant que serveur web pour héberger la page de configuration, afin de permettre à l'utilisateur d'accéder à l'interface de configuration, pour saisir les paramètres WiFi et les informations utilisateur (e-mail et mot de passe).

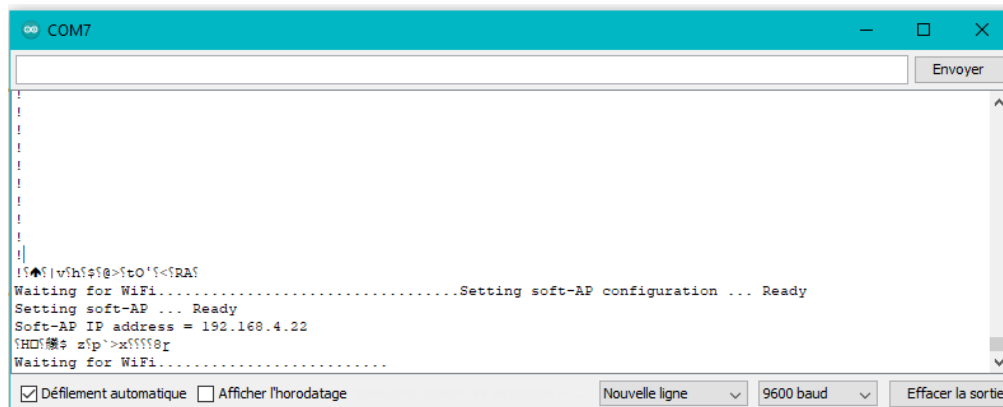
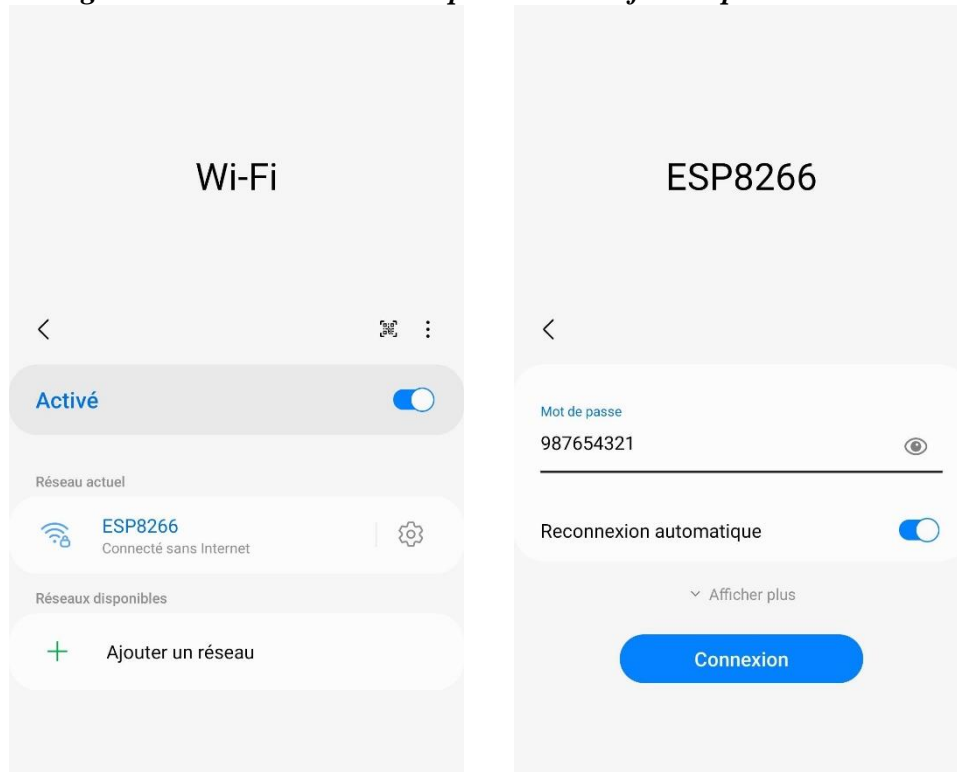


Figure III.15 : Affichage des paramètres de point d'accès sur le moniteur série



**Figure III.16 : Connexion au point d'accès fourni par le WEMOS**

Pour accéder à l'interface de configuration, il suffit de se rendre à l'adresse IP "192.168.4.22" sur le navigateur, cette adresse est statique (prédéfinie dans le code).

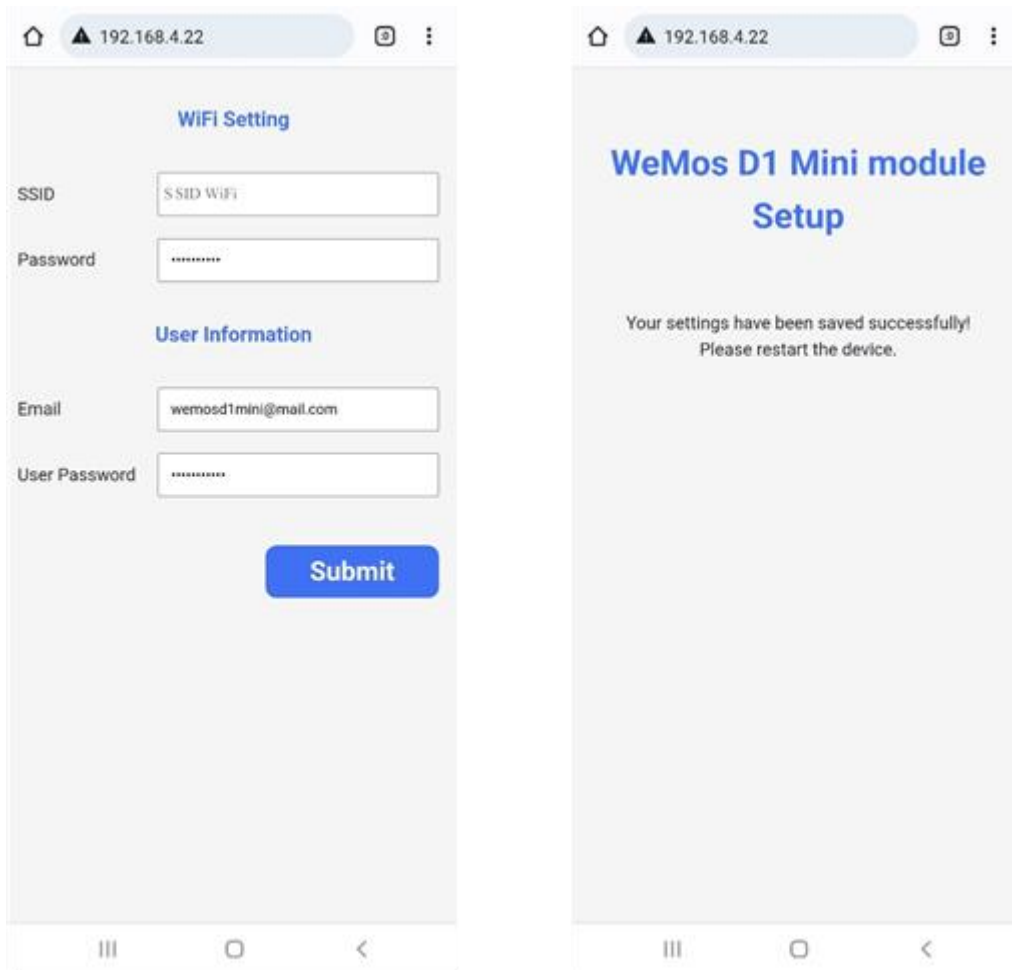


Figure III.17 : Interface de configuration du WEMOS

Résultats de la fonction "setup" :

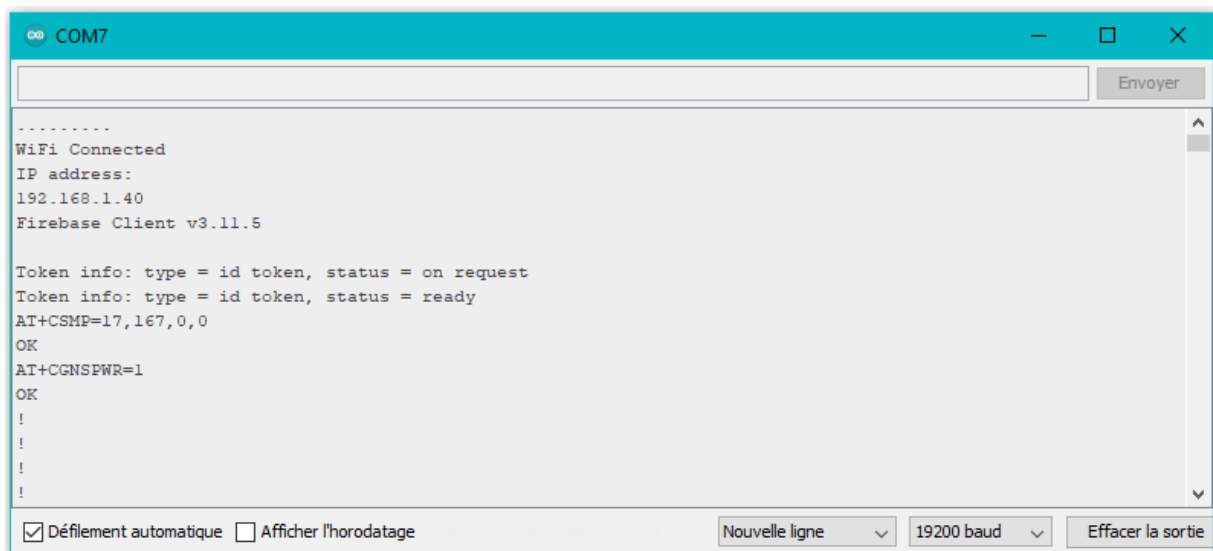


Figure III.18 : Résultats de la fonction "setup" affichés sur le moniteur série

La représentation du signal ECG sur le traceur série est donnée par la Figure III.19 :

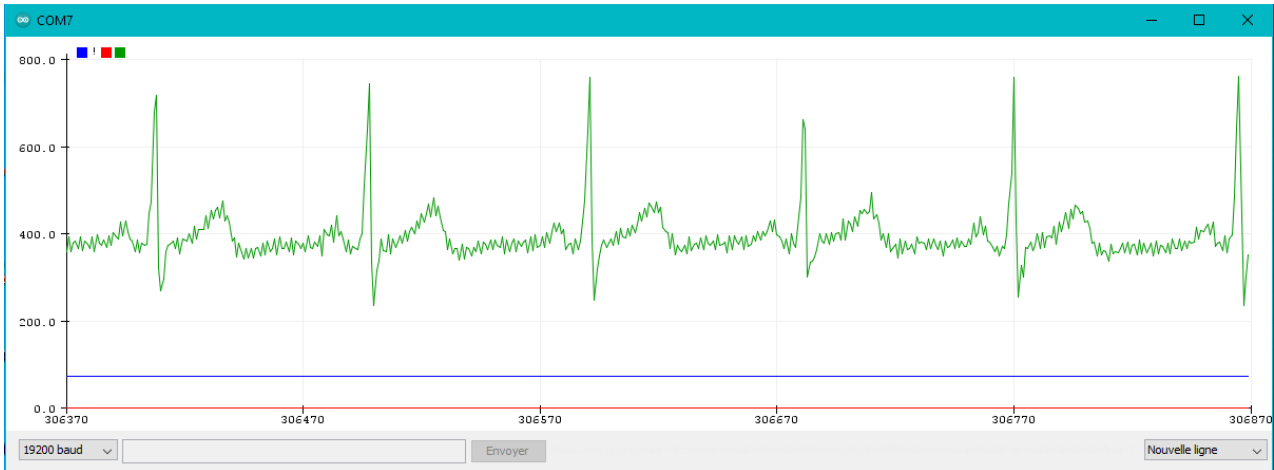


Figure III.19 : Lecteur du signal ECG sur le traceur série

Les valeurs d'ECG stockées dans Firebase sont illustrées par la Figure III.20 :

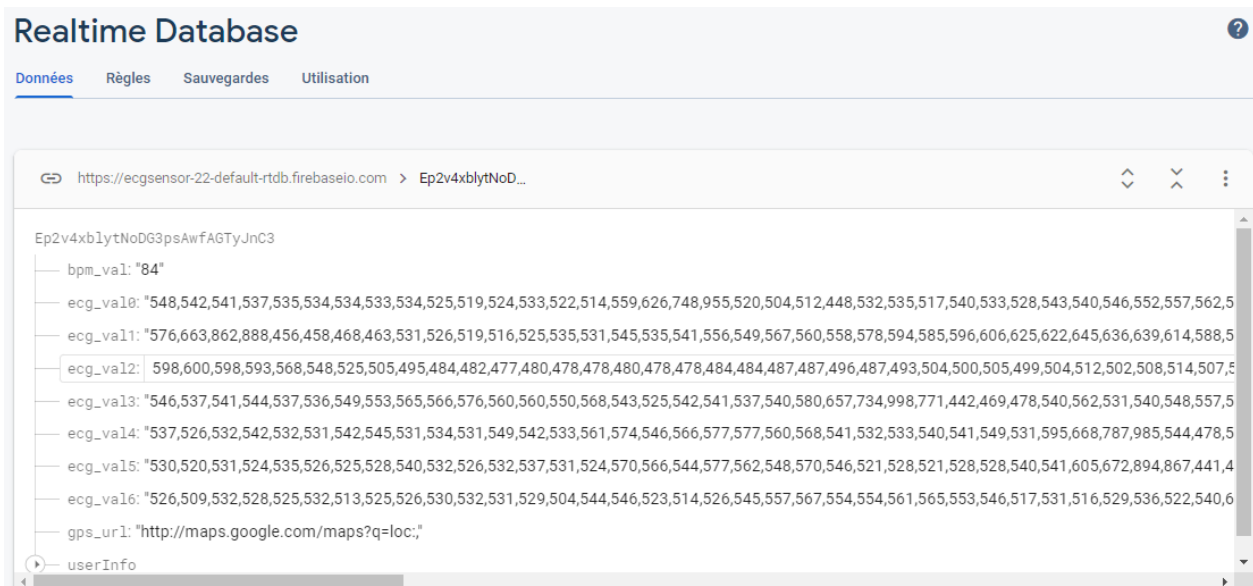


Figure III.20 : Transfert des données en temps réel vers Firebase

La figures III.21 et la figures III.22 représentent les interfaces utilisateur de l'application web. Une fois l'authentification est réussie, l'utilisateur sera redirigé vers la page de surveillance sur laquelle il peut consulter le signal ECG et la valeur du BPM.

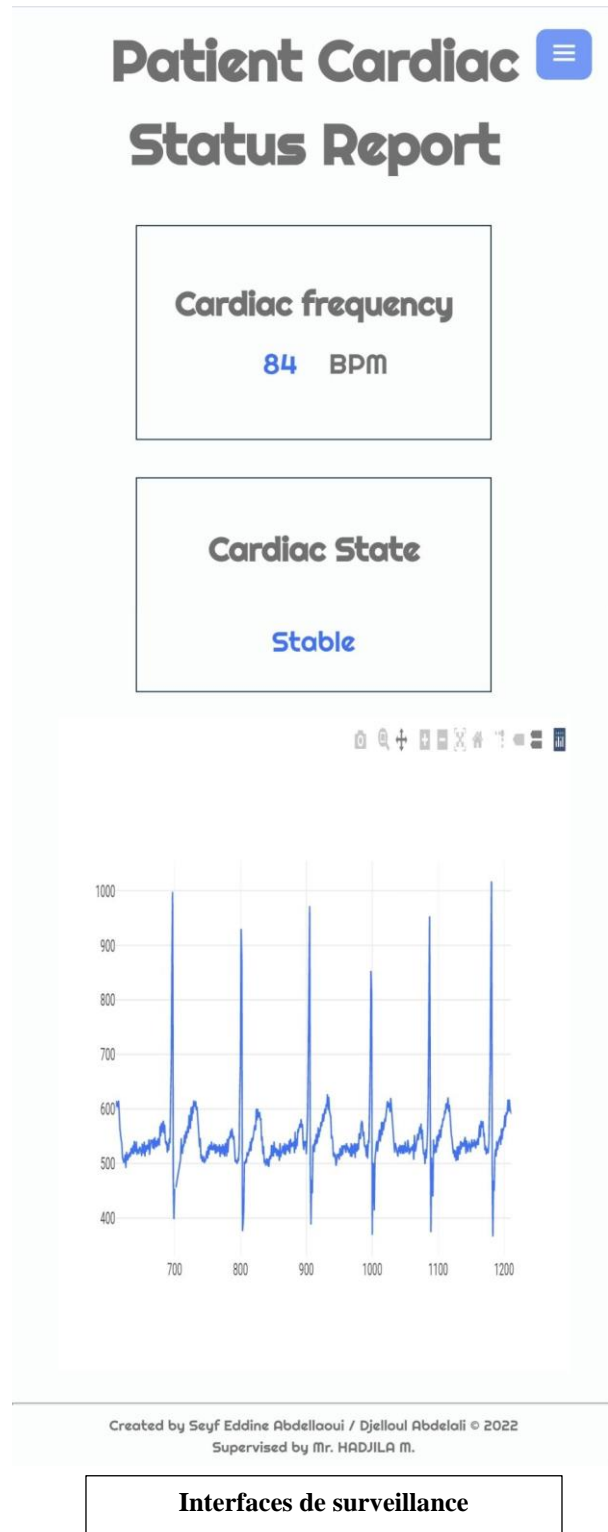
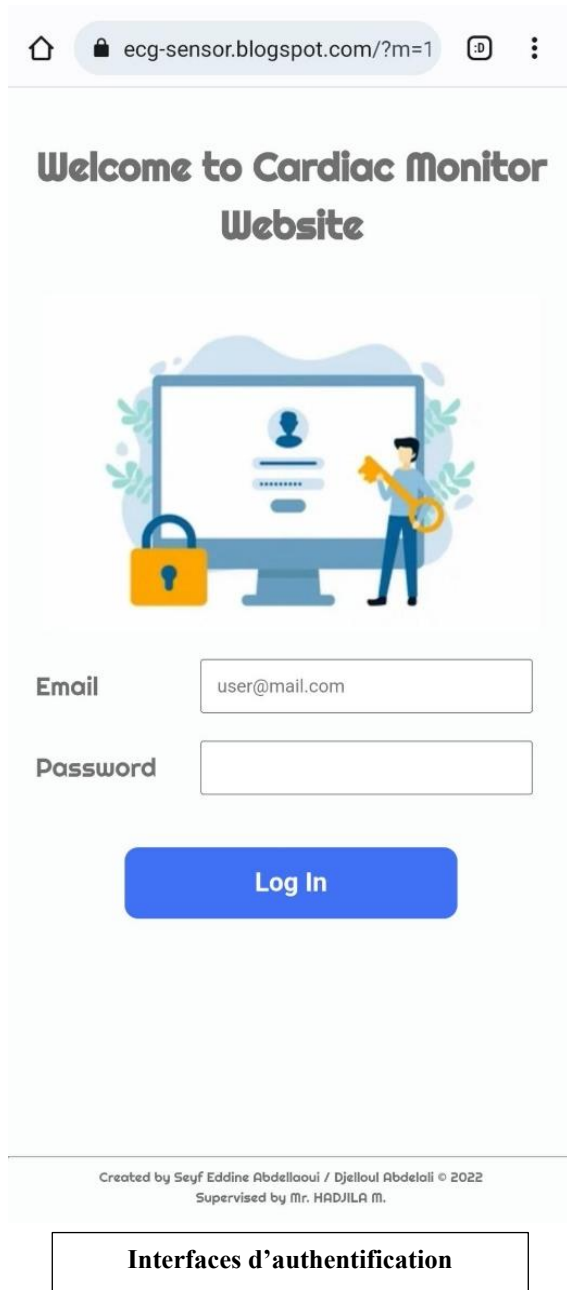
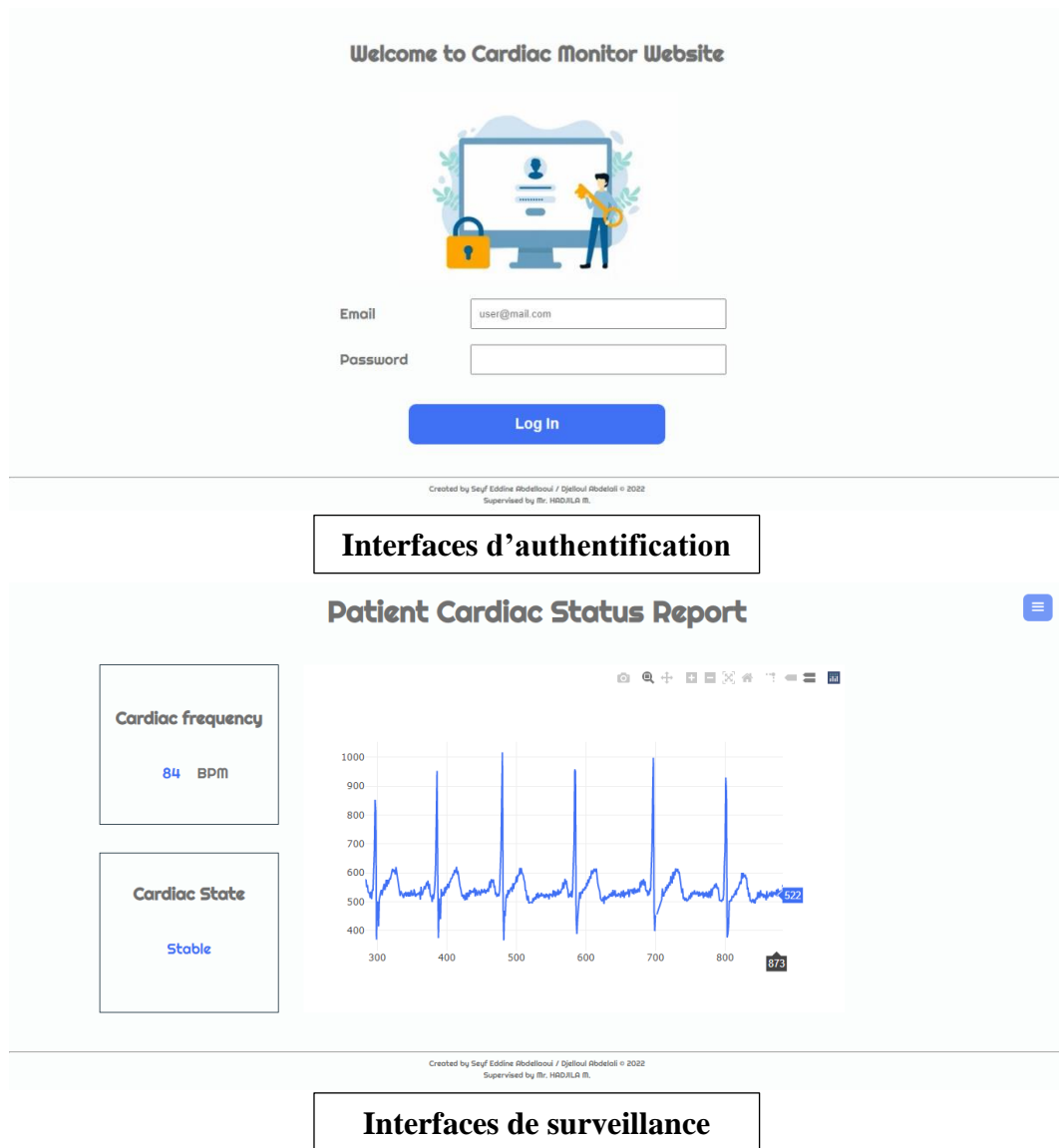
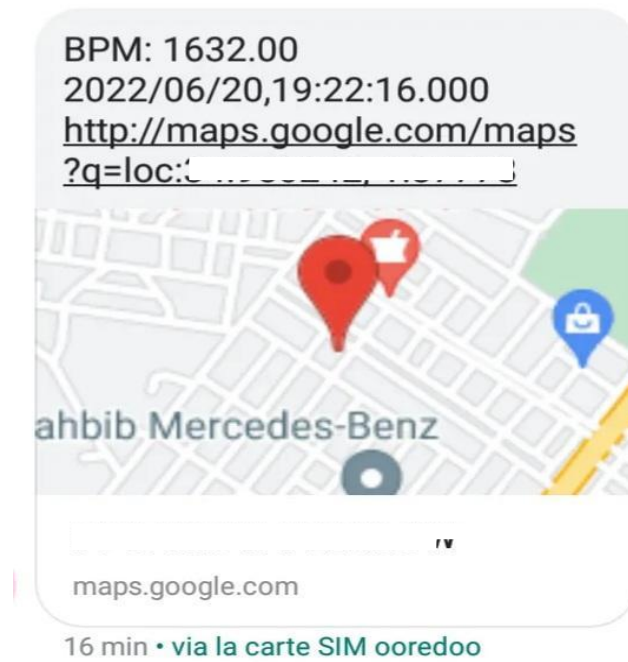


Figure III.21 : Interface utilisateur de l'application web version mobile



**Figure III.22 : Interfaces utilisateur de l'application web version PC**

Afin d'obtenir une valeur BPM anormale, nous avons fait du bruit sur le capteur ECG, et par conséquent notre système a envoyé une alerte par SMS pour informer le service de surveillance de l'état du patient. Ce SMS comprend le nombre de battements par minute (BPM), la date (AAAA/MM/JJ), l'heure (hh:mm:ss) et un lien vers la position GPS sur Google Maps, comme illustré à la Figure III.23.



*Figure III.23 : Alerte par SMS*

### III.11. Conclusion

Dans ce chapitre, nous avons décrit le travail pratique de la réalisation du système de surveillance basé sur l'Internet des Objets, et nous avons présenté les résultats obtenus.

Dans tout le travail pratique, nous avons trouvé toujours des problèmes tels que la lecture et l'envoi du signal ECG capté à la base de données en temps réel sans le déformer tout en minimisant les pertes d'informations possibles. Ce problème a été résolu par la création de la fonction `sendData()`.

# *Conclusion Générale*

## *Conclusion Générale*

---

Dans ce projet de fin d'études, nous avons conçu et réalisé un système à base d'IoT afin de répondre aux besoins de la télésurveillance pour les patients souffrant d'une maladie cardiaque. Pour ce genre de maladie, le signal ECG constitue un élément essentiel pour déterminer les anomalies cardiaques.

En réalisant ce projet, nous avons eu l'opportunité de rencontrer et d'apprendre énormément de choses telles que la programmation Arduino, la manipulation et l'utilisation de composants (ESP8266, SIM808) et de découvrir le domaine médical notamment la télésurveillance médicale. Cela nous a également permis l'ouverture d'un champ d'applications et de recherche relativement important.

Le travail réalisé peut être amélioré, de plusieurs côtés, pour rendre le système plus simple et plus flexible en ajoutant d'autres fonctionnalités, par exemple : développer l'application web ou la convertir en application Android et la rendre multi-utilisateurs pour suivre autant de patients que possible, et des fonctions telles que la prise de rendez-vous et d'autres qui facilitent les transactions entre le patient et la clinique. D'autre part, on peut diviser le système en plusieurs nœuds et ajouter des autres capteurs médicaux afin de le rendre plus intelligent.

Nous souhaitons que ce modeste travail servira davantage pour les promotions qui viennent et de leur donner une image réelle sur l'importance de la télésurveillance médicale.



## *Bibliographie*

- [1] Billet, B. (2015). Système de gestion de flux pour l'Internet des objets intelligents (Doctoral dissertation, Université de Versailles-Saint Quentin en Yvelines).
- [2] Sharma, C., Bairwa, H., Pareek, K., & Joshi, N. (2021). Automatic Ration Management System, International journal of Engineering, Business and Management (IJEEM).
- [3] <https://fr.statista.com/statistiques/584481/internet-des-objets-nombre-d-appareils-connectes-dans-le-monde--2020> [Consulté en Février 2022]
- [4] Ngu, A. H., Gutierrez, M., Metsis, V., Nepal, S., & Sheng, Q. Z. (2016). IoT middleware: A survey on issues and enabling technologies. IEEE Internet of Things Journal, 4(1), 1-20.
- [5] Benghozi, P. J., Bureau, S., & Massit-Folléa, F. (2015). L'Internet des objets/The Internet of Things: Quels enjeux pour l'Europe?/What Challenges for Europe?. Les Editions de la MSH.
- [6] Tebib, S., & Abdelghani, S. (2017). La Sécurité Des Données Dans L'Internet Des Objets (IoT) (Doctoral dissertation, Université laarbi tebessi tebessa).
- [7] Tomar, A. (2011). Introduction to ZigBee technology. Global Technology Centre, 1, 1-24.
- [8] <https://www.tactis.fr/reseaux-cellulaires-mobiles/> [Consulté en Février 2022]
- [9] <https://www.objetconnecte.com/tout-savoir-sur-sigfox/> [Consulté en Février 2022]
- [10] <https://www.tactis.fr/smart-city-exemples-france/> [Consulté en Février 2022]
- [11] <https://www.picomto.com/pourquoi-integrer-de-liot-dans-lindustrie/> [Consulté en Mars 2022]
- [12] <https://blog.hellocare.pro/objets-connectes-sante> [Consulté en Mars 2022]
- [13] <https://www.synox.io/votre-secteur/sante-connectee/> [Consulté en Mars 2022]
- [14] Saleh, I. (2017). Les enjeux et les défis de l'internet des objets. In Conférence "Nouveaux défis de l'Internet des Objets (IdO).
- [15] <https://easypartner.fr/blog/comment-renforcer-la-securite-des-donnees-dans-IIoT-%F0%9F%94%92/> [Consulté en Avril 2022]
- [16] <https://www.crowdstrike.fr/cybersecurity-101/cloud-security/> [Consulté en Avril 2022]
- [17] El Jaouhari, S., Bouabdallah, A., Bonnin, J. M., & de Rennes-France, S. (2016). La sécurité des objets connectés. MISC: multi-system & internet security cookbook, (88), 54-59.
- [18] <https://iotindustriel.com/tendances-de-liot-industriel/web-of-things-cest-quoi> [Consulté en Avril 2022]
- [19] Hakima, T., & Ferroudja, S. (2016). Conception et réalisation d'un système antivol à base d'une carte Arduino (Doctoral dissertation, Université Mouloud Mammeri).

## *Bibliographie*

---

[20] Hughes, J. M. (2016). *Arduino: a technical reference: a handbook for technicians, engineers, and makers.* " O'Reilly Media, Inc."

[21] Djafaar, R. H. (2019). *Penunjuk Arah Kiblat dan Waktu Shalat Menggunakan Global Positioning System (GPS) Berbasis Wemos D1 Mini* (Doctoral dissertation, Institut Teknologi Sepuluh Nopember).

[22] <https://www.simcom.com/product/SIM808.html> [Consulté en Avril 2022]

[23] Beya, O. (2014). *Analyse et reconnaissance de signaux vibratoires: contribution au traitement et à l'analyse de signaux cardiaques pour la télémédecine* (Doctoral dissertation, Dijon).

[24] BENRETEM, Z. (2021). *Exploitation des séries chronologiques pour l'étude des données évolutives.*

[25] Sabrina, B., & Fadhila, H. (2015). *Implémentation d'algorithmes DSP sur la carte Arduino pour un signal ECG* (Doctoral dissertation, Université Mouloud Mammeri).

[26] Lakhdari, S., Meziani, A., & Rechem, D. (2020). *Etude et réalisation d'un système cardio-fréquence-mètre en temps réel pour smartphone.*

[27] Kanani, P., & Padole, M. *IoT based Eye Movement Guided Wheelchair driving control using AD8232 ECG Sensor.* *International Journal of Recent Technology and Engineering*, 8(4), 5013-5017.

[28] Hamad, A. M., & Jasim, A. D. (2021). *Remote ECG signal monitoring and classification based on Arduino with AD8232 sensor.* *University of Thi-Qar Journal for Engineering Sciences*, 11(2), 95-101.

[29] Hasan, D., & Ismaeel, A. (2020). *Designing ECG monitoring healthcare system based on internet of things blynk application.* *Journal of applied science and technology trends*, 1(3), 106-111.

[30] Kartit, Z. (2016). *Contribution à la sécurité du Cloud Computing: Application des algorithmes de chiffrement pour sécuriser les données dans le Cloud Storage.*

[31] Medhioub, H. (2015). *Architectures et mécanismes de fédération dans les environnements Cloud Computing et Cloud Networking* (Doctoral dissertation, Evry, Institut national des télécommunications).

# Annexe

```
/******  
Librairies  
*****/  
#include <EEPROM.h>  
#include <ESP8266WiFi.h>  
#include <ESP8266WebServer.h>  
#include <SoftwareSerial.h>  
#include <FirebaseESP8266.h>  
#include <addons/TokenHelper.h>// Provide the token generation process  
info.  
#include <addons/RTDBHelper.h>// Provide the RTDB payload printing  
info and other helper functions.  
#include "setupHtmlPage.h";  
/******  
Variables  
*****/  
#define PulseSensorPin A0 // Pulse Sensor PURPLE WIRE  
connected to ANALOG PIN 0  
#define LED7 D7 // The on-board Arduion LED  
#define LED8 D8 // The on-board Arduion LED  
#define LO1 D5 // LO+  
#define LO2 D6 // LO-  
  
#define PIN_TX D2  
#define PIN_RX D1  
  
#define DATABASE_URL "ecgsensor-22-default-rtdb.firebaseio.com"  
#define API_KEY "AIzaSyDleaZgb6wfa4i2iWsSLNikuGSRwAn2sUA"  
  
#define PHONE_NUMBER "(000)0000000"  
/******  
Variables  
*****/  
String userId;  
int Signal; // holds the incoming raw data. Signal  
value can range from 0-1024  
int Threshold = 700; // = 516; // Determine which Signal to  
"count as a beat", and which to ingore.  
  
long instancel = 0, timer;  
double hrv = 0, hr = 72, interval = 0;  
int count = 0;  
bool flag = 0;  
// Space to store values to send  
String data[5];  
String state,timegps,latitude,longitude;  
  
int j = 0, x = 0;  
String str_ECG = "", str_GPS="http://maps.google.com/maps?q=loc:";  
String ecgTab[8];
```

```
int tries = 0;
struct settings {
  char ssid[30];
  char password[30];
  char userEmail[30];
  char userPass[30];
}
userInfo = {};
const char *ssid = "ESP8266";
const char *password = "987654321";

IPAddress local_IP(192,168,4,22);
IPAddress gateway(192,168,4,9);
IPAddress subnet(255,255,255,0);
/*****
                Objectes
*****/
ESP8266WebServer server(80);
SoftwareSerial sim808(PIN_TX, PIN_RX); //Connect RX,TX,PWR
// Define Firebase Data object
FirebaseData fbdo;
FirebaseAuth auth;
FirebaseConfig FBconfig;
/*****
                Functions
*****/

void setup() {
  EEPROM.begin(sizeof(struct settings));EEPROM.get(0, userInfo);
  // initialize the serial communication:
  Serial.begin(19200);
  // connect to wifi.
  WiFi.mode(WIFI_STA);
  // WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  WiFi.begin(userInfo.ssid, userInfo.password);
  sim808.begin(19200);
  // Assign the pin as INPUT
  pinMode(PulseSensorPin, INPUT); // Setup
  pinMode(LED7,OUTPUT); // pin that will blink to your heartbeat!
  pinMode(LED8,OUTPUT); // GREEN LED!
  pinMode(LO1, INPUT); // Setup for leads off detection LO +
  pinMode(LO2, INPUT); // Setup for leads off detection LO -
  Serial.println();
  Serial.print("Waiting for WiFi...");
  // Wait for connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);Serial.print(".");
    if (tries++ == 30) {
      WiFi.disconnect();
      WiFi.mode(WIFI_AP);
      Serial.println();
      Serial.print("Setting soft-AP configuration ...
");Serial.println(WiFi.softAPConfig(local_IP, gateway, subnet) ?
"Ready" : "Failed!");
```

```
    Serial.print("Setting soft-AP ...
");Serial.println(WiFi.softAP(ssid,password) ? "Ready" : "Failed!");
    Serial.print("Soft-AP IP address =
");Serial.println(WiFi.softAPIP());
    server.on("/", handleSetupPage);
    server.begin();
}
while (tries > 30) {server.handleClient();}
}
Serial.println("");
Serial.println("WiFi Connected");
Serial.println("IP address: ");
Serial.println(WiFi.localIP());
Serial.printf("Firebase Client v%s\n\n", FIREBASE_CLIENT_VERSION);
auth.user.email = userInfo.userEmail; //Assign the user sign in
credentials
auth.user.password = userInfo.userPass;
FBconfig.api_key = API_KEY; //Assign the api key (required)
FBconfig.database_url = DATABASE_URL; // Assign the RTDB URL
(required)
/* Assign the callback function for the long running token
generation task */
FBconfig.token_status_callback = tokenStatusCallback; // see
addons/TokenHelper.h
Firebase.begin(&FBconfig, &auth);
Firebase.setDoubleDigits(5); delay(50);
userId = auth.token.uid.c_str(); delay(500);
// sim808 setup
sim808Data("AT+CSMP=17,167,0,0"); delay(100); // set this parameter
if empty SMS received
sim808Data("AT+CGNSPWR=1"); delay(50);
sim808Data("AT+CGNSSEQ=RMC"); delay(150);
}

void loop() {
if((digitalRead(LO1) == 1)|| (digitalRead(LO2) == 1)){
digitalWrite(LED8,LOW);
Serial.println('!');
instance1 = micros();
timer = millis();
} else {
digitalWrite(LED8,HIGH);
readHeartBeat();
sendData(Signal, hr);
if ((hr < 60) || (hr > 100)) {
digitalWrite(LED7,HIGH);
//if (x > 30000) x = 0;
sendGPSbySMS();
}
//x++;
Serial.print(hr);
Serial.print(",");
Serial.print(hrv);
Serial.print(",");
Serial.println(Signal);
}
}
```

---

```
}
void readHeartBeat() {
    Signal = analogRead(PulseSensorPin); //Read the PulseSensor's value.
    //Signal = map(Signal, 250, 400, 0, 100); //to flatten the ecg
values a bit
    if ((Signal > Threshold) && (!flag)){ // If the signal is
above "550", then "turn-on" Arduino's on-Board LED.
        digitalWrite(LED7,HIGH);
        count++;
        flag = 1;
        interval = micros() - instancel; //RR interval
        instancel = micros();
    } else if (Signal < Threshold) {
        digitalWrite(LED7,LOW); // Else, the sigal must be below
"550", so "turn-off" this LED.
        flag = 0;
    }
    if ((millis() - timer) > 10000) {
        hr = count*6;
        timer = millis();
        count = 0;
    }
    hrv = hr/60 - interval/1000000;
}

void sendData(int ECG, int BPM){
    if ((millis() - timer) < 9950) {
        if (j < 7) {
            str_ECG = str_ECG + ECG + "," ; // Adds the value
            if (str_ECG.length() >= 400){
                ecgTab[j] = str_ECG;
                str_ECG = ""; j++;
            }
        }
    }
    if ((WiFi.status() == WL_CONNECTED)&&(Firebase.ready())){
        T++;
        if (T > 8640) T = 0;
        String str1 = userId + "/" + T + "/bpm_val";
        String str2 = userId + "/" + T + "/gps_url";
        String str3 = userId + "/T";
        Firebase.setString(fbdo, str2, str_GPS);
        Firebase.setInt(fbdo, str1, BPM);
        Firebase.setInt(fbdo, str3, T);
        j = 0;
        for (int n = 0; n < 7; n++){
            String str = userId + "/" + T + "/ecg_val" + n;
            Firebase.setString(fbdo, String(str), ecgTab[n]);
        }
        // handle error
        if ((!Firebase.setString(fbdo, str2, str_GPS)) ||
(!Firebase.setString(fbdo, str1, BPM))) {
            Serial.print("setting /message failed:");
            Serial.println(fbdo.errorReason());
            return;
        }
    }
}
}
```

---

```
void sim808Data(String command) {
  sim808.println(command);
  while (sim808.available()) {
    char r = sim808.read();
    if (r != -1) Serial.write((char) r);
  } delay(50);
}
void sendGPSbySMS() {
  if ((millis() - timer) < 3000) { // 3 seconds timer to calculate hr
    //sim808Data("AT+CGNSINF"); delay(100);
    sim808.print("AT+CGNSINF"); delay(50);
    int i = 0, d = 1;
    while (sim808.available() && i < 5){
      char c = sim808.read();
      if (c != ',') {
        data[i] += c;
        if (i == 2){
          if (d == 4 || d == 6){ data[2] += "/"; }
          if (d == 8){ data[2] += ","; }
          if (d == 10 || d == 12){ data[2] += ":"; }
          d++;
        }
        } else { i++; data[i] = "";}
      }
    state = data[1];
    timegps = data[2];
    latitude = data[3];
    longitude = data[4];
    str_GPS = "BPM: " + String(hr) + " " + timegps + "
http://maps.google.com/maps?q=loc:"+latitude+","+longitude;
  } else if ((millis() - timer) > 9500 && state != 0 && x == 0) {
    Serial.println("Sending SMS..."); //Show this message on serial
monitor
    sim808.print("AT+CMGF=1\r");delay(100); //Set the module to SMS
mode
    sim808.print("AT+CMGS=\""); delay(500);
    sim808.print(PHONE_NUMBER);
    sim808.println("\r"); delay(500);
    sim808.print(str_GPS); delay(500); //This is the text to send to the
phone number, don't make it too long or you have to modify the SoftwareSerial
buffer
    sim808.print((char)26); delay(500); // ||sim808.write(0x1A); // End AT
command with a ^Z, ASCII code 26
    sim808.println();
    Serial.println(str_GPS);
    Serial.println("Text Sent.");
    delay(500);
  }
}
void handleSetupPage() {
  String setupPage;
  if (server.method() == HTTP_POST) {
    setupPage = setupHtml1;
    String p1 = server.arg("ssid");String p2 = server.arg("password");
    String p4 = server.arg("userEmail");String p5 = server.arg("userPass");
    strncpy(userInfo.ssid, p1.c_str(), sizeof(userInfo.ssid)
);strncpy(userInfo.password, p2.c_str(), sizeof(userInfo.password) );
    strncpy(userInfo.userEmail, p4.c_str(), sizeof(userInfo.userEmail)
);strncpy(userInfo.userPass, p5.c_str(), sizeof(userInfo.userPass) );
    userInfo.ssid[p1.length()] = userInfo.password[p2.length()] =
userInfo.userEmail[p4.length()] = userInfo.userPass[p5.length()] = '\0';
    EEPROM.put(0, userInfo); EEPROM.commit();
  }
}
```

```
server.send(200, "text/html", setupPage);
} else {
    setupPage = setupHtml;
    setupPage.replace("@@ssid@",
userInfo.ssid);setupPage.replace("@@password@", userInfo.password);
    setupPage.replace("@@userEmail@",
userInfo.userEmail);setupPage.replace("@@userPass@", userInfo.userPass);
    server.send(200, "text/html", setupPage);
}
}
```