

N° Ordre : ...

# Mémoire de Projet de Fin d'Etudes

Présenté pour obtenir  
La Licence Professionnelle en :  
Technologue en Instrumentation et Maintenance Biomédicale

## Développement d'une Application IoT pour la Surveillance médicale à distance

Réalisé par :

Mr. **RHIBA** Zakaria

Mr. **TAKILI** Youssef

Encadré Par :

Pr. **ELHANINE MUSTAPHA**

*Soutenu publiquement le /06/2022*

*Devant le jury :*

Pr.

Pr.

Pr.

Année universitaire : 2022/2023

# Remerciements

*Nous tenons à la fin de ce travail à remercier ALLAH le tout puissant de nous avoir donné la foi et de nous avoir permis d'en arriver là.*

*Nos vifs remerciements accompagnés de toute notre gratitude vont également à notre encadrant Pr Mustapha EL HANINE aussi à notre encadrant extérieur Mr MOCHTARAY. Nous vous sommes particulièrement reconnaissante pour se soutiennent, ses disponibilités, ses précieux conseils avisés, ses orientations et ses encouragements*

*Nos remerciements vont également à nos parents, à tous les membres de nos familles pour tous les sacrifices qu'ils ont consentis pour nous permettre de suivre notre étude dans les meilleures conditions possibles et n'ont jamais cessé de nous encourager tout au long de nos années d'études*

*Nous remercierons les membres de jury de m'avoir fait l'honneur d'accepter de participer à mon jury. Je remercie ceux qui m'ont aidé à l'aboutissement de ce travail.*

*Nos formidables promotions 2023 du licence professionnelle technologue en instrumentation et maintenance biomédicale de l'institut supérieur des sciences de la santé de settat et aussi à toute l'équipe de service biomédical de CHUIR.*

***MERCI A TOUT.***

## Résumé

Dans le cadre de ce projet de fin d'études, nous avons développé une application desktop innovante utilisant PyQt5 en Python. Cette application offre la possibilité de surveiller à distance les patients et d'étudier leur évolution dans un contexte plus général, au-delà de la réanimation.

L'application offre une interface conviviale et intuitive, permettant aux professionnels de la santé de visualiser en temps réel les données des patients et de surveiller leur évolution de manière précise. De plus, des seuils prédéfinis sont configurés pour chaque paramètre surveillé. Si ces seuils sont atteints, des alertes sont automatiquement envoyées par email au médecin responsable, l'informant de la situation critique du patient.

Cette solution a pour objectif d'aider les professionnels de la santé à prendre des décisions éclairées et précises en fournissant des données en temps réel, même en dehors de la présence physique du patient. Elle facilite la surveillance à distance et contribue à une prise en charge proactive des patients.

Grâce à notre application desktop et au petit moniteur développé, nous espérons améliorer la qualité des soins médicaux en offrant une solution efficace et pratique pour la surveillance à distance des patients.

**Mots clés**— Surveillance à distance des patients, Solution de surveillance, protocoles de communication, Protocole MQTT, Données en temps réel

## **Abstract**

As part of this final year project, we have developed an innovative desktop application using PyQt5 in Python. This application provides the ability to remotely monitor and study patients parameters progress in a more general context, beyond just the intensive care unit.

We have also designed a small monitor using the ESP8266 microcontroller, capable of measuring essential data such as oxygen saturation ( $\text{SpO}_2$ ), temperature, and heart rate. This monitor transmits this data via the MQTT protocol to the desktop application.

The application offers a user-friendly and intuitive interface, allowing healthcare professionals to view real-time patient data and monitor their progress accurately. Additionally, predefined thresholds are set for each monitored parameter. If these thresholds are reached, automatic email alerts are sent to the responsible doctor, informing them of the critical condition of the patient.

The goal of this solution is to assist healthcare professionals in making informed and precise decisions by providing real-time data, even when they are not physically present with the patient. It facilitates remote monitoring and contributes to proactive patient care.

With our developed desktop application and small monitor, we aim to enhance the quality of medical care by providing an efficient and convenient solution for remote patient monitoring.

**Keywords**— Remote patient monitoring, Monitoring solution, MQTT protocol, Real-time data

# List des abréviations

**API** application programming interface

**CHUIR** Centre hospitalier universitaire Ibn Rochd

**COAP** Constrained Application Protocol

**HTTP** Hypertext Transfer Protocol

**IdO** Internet des Objets

**IdOM** Internet des objets médicaux

**IoT** Internet of Things

**MQTT** Message Queuing Telemetry Transport

**PCB** Printed circuit board

# Table des matières

<b>Table des figures</b>	iv
<b>Liste des tableaux</b>	vi
<b>Introduction</b>	1
<b>1 Présentation du lieu de stage et Problématique</b>	2
1.1 Présentation du Centre Hospitalier Universitaire Ibn Rochd . . . . .	2
1.1.1 Historique . . . . .	3
1.1.2 Organigramme de CHUIR . . . . .	4
1.1.3 La présentation du service biomédical . . . . .	4
1.2 Les taches effectuées durant le stage . . . . .	5
1.3 Problématiques observées . . . . .	6
<b>2 La surveillance médicale à distance à base d'internet des objets médicaux</b>	8
2.1 L'internet des objets . . . . .	8
2.1.1 Définition . . . . .	9
2.1.2 Les différents processus de l'IoT . . . . .	9
2.1.3 Les objets Connectés . . . . .	9
2.1.4 Les avantages de l'IoT . . . . .	11
2.1.5 Les inconvénients de l'IoT . . . . .	11
2.1.6 La sécurité dans l'IoT . . . . .	11
2.2 L'internet des objets médicaux . . . . .	12
2.2.1 La surveillance médicale . . . . .	12
2.2.2 Types de surveillance médicale . . . . .	13
2.2.3 Définition de l'IdOM . . . . .	13
2.2.4 Avantages et inconvénients d'utilisation de technologie IdOM . . . . .	14
2.2.5 Rôle de l'IdOM dans le domaine de la santé . . . . .	14
2.3 Architecture et Protocoles dans IdO et IdOM . . . . .	15
2.3.1 Architecture de réseau IdOM . . . . .	15
2.3.2 Les protocoles de communication : . . . . .	16
<b>3 Méthodes de réalisation et Implémentation du prototype</b>	18
3.1 Planification du projet . . . . .	18
3.2 Méthodes de réalisation matérielle . . . . .	19
3.2.1 Les Capteurs . . . . .	19
3.2.2 Les Microcontrôleurs . . . . .	22
3.2.3 Interface de sortie . . . . .	23

3.3	Méthodes de réalisation Logicielle . . . . .	26
3.3.1	Python : . . . . .	26
3.3.2	Qt designer : . . . . .	26
3.3.3	PyQt5 : . . . . .	27
3.3.4	Pandas : . . . . .	27
3.3.5	VS Code : . . . . .	28
3.3.6	Arduino IDE : . . . . .	28
3.4	Développement des maquettes de l'application . . . . .	29
3.5	Développement de Moniteur . . . . .	35
3.6	Connectivité MQTT . . . . .	37
3.7	Concevoir la disposition du PCB . . . . .	39
3.8	Conception De Boîtier . . . . .	42
	<b>Conclusion</b>	<b>45</b>
	<b>Bibliographie</b>	<b>46</b>

# Table des figures

1.1	Accueil Ibn Rochd . . . . .	2
1.2	Etablissements médicaux Ibno Rochd . . . . .	3
1.3	Organigramme de l'hôpital . . . . .	4
1.4	Salle de réanimation . . . . .	6
2.1	Internet des objets . . . . .	8
2.2	Architecture d'un objet connecté . . . . .	9
2.3	Internet des objets médicaux . . . . .	12
2.4	Internet des objets médicaux options . . . . .	14
2.5	Architecture de réseau IdOM . . . . .	15
2.6	protocole HTTP . . . . .	16
2.7	protocole MQTT . . . . .	17
2.8	COAP Diagrama . . . . .	17
3.1	Diagramme de GANTT . . . . .	18
3.2	Capteur de température analogique LM35 . . . . .	19
3.3	Oxymètre MAX30100 . . . . .	20
3.4	Schéma fonctionnel du système MAX30100 . . . . .	20
3.5	NodeMCU ESP8266 . . . . .	22
3.6	Arduino Uno . . . . .	23
3.7	LCD I2C . . . . .	24
3.8	BUZZER . . . . .	25
3.9	LED's . . . . .	25
3.10	Python . . . . .	26
3.11	Qt designer . . . . .	26
3.12	PyQt5 . . . . .	27
3.13	Pandas . . . . .	28
3.14	VS Code . . . . .	28
3.15	Arduino IDE . . . . .	29
3.16	Home Interface . . . . .	30
3.17	Interface Principale . . . . .	30
3.18	Interface de connexion . . . . .	31
3.19	Interface des Patients . . . . .	32
3.20	Interface de modification . . . . .	32
3.21	Interface de consultation . . . . .	33
3.22	Interface de supervision . . . . .	33
3.23	CSV . . . . .	34

3.24 Tableau base de données 1 . . . . .	35
3.25 Tableau base de données 2 . . . . .	35
3.26 connexions entre max30100 et l'ESP8266 . . . . .	36
3.27 connexions entre lm35 et l'ESP8266 . . . . .	36
3.28 code de connexion C/C++ mqtt . . . . .	37
3.29 code de connexion python mqtt . . . . .	38
3.30 Architecture MQTT . . . . .	39
3.31 platine de prototypage . . . . .	40
3.32 Schéma de circuit . . . . .	40
3.33 PCB à imprimer . . . . .	41
3.34 PCB 3D . . . . .	41
3.35 résultat final du boîtier de moniteur . . . . .	43
3.36 intérieur du boîtier de moniteur . . . . .	43

# Liste des tableaux

1.1 Fiche Signalétique . . . . .	3
1.2 Les taches effectuées durant le stage . . . . .	6

# Introduction

La surveillance médicale des patients en milieu hospitalier, en particulier dans les salles de réanimation, est d'une importance vitale pour assurer des soins de qualité et des interventions rapides. Cependant, le nombre limité de lits médicaux et de moniteurs disponibles peut entraver cette surveillance continue et compromettre la prise en charge des patients. Face à cette problématique, notre projet de stage se concentre sur le développement d'une solution innovante pour le suivi à distance des patients, afin de pallier ces contraintes et d'améliorer la qualité des soins.

L'objectif principal de notre projet est de concevoir et de développer une application IoT permettant de surveiller à distance l'état des patients. Nous cherchons à créer une solution portable et pratique, utilisant des technologies telles que les capteurs de saturation en oxygène ( $\text{SpO}_2$ ), de température et de fréquence cardiaque et tout autre capteur selon les paramètres souhaités à supervisé. L'application sera capable de collecter, de transmettre et d'afficher ces données vitales en temps réel, offrant ainsi une surveillance continue en temps réel et efficace des patients, même en l'absence de lits médicaux et de moniteurs disponibles, ainsi le patient peut installer chez lui.

Pour atteindre nos objectifs, nous avons adopté une approche multidisciplinaire combinant les domaines de la biomédical, de l'informatique et de l'ingénierie des systèmes embarqués. Nous avons utilisé le microcontrôleur ESP8266 pour développer un moniteur portable capable de collecter les données des capteurs. Ces données sont ensuite transmises via le protocole MQTT à une plateforme centrale, où elles sont traitées et affichées à l'aide d'une application desktop. Nous avons utilisé PyQt5, une bibliothèque Python, pour développer une interface conviviale permettant aux professionnels de la santé de visualiser les données des patients et de recevoir des alertes en cas de variations critiques.

Ce projet de stage vise à apporter plusieurs contributions significatives dans le domaine de la surveillance médicale à distance. En développant cette solution IoT, nous espérons offrir une alternative pratique et précise à la surveillance traditionnelle des patients. En permettant aux médecins et aux infirmières de surveiller les signes vitaux des patients à distance, nous visons à améliorer la qualité des soins, à optimiser l'utilisation des ressources médicales et à faciliter la prise de décisions éclairées.

Ce rapport contiendra la synthèse de notre activité pendant ce Stage, la présentation suivante exposera les divers axes, le manuscrit est organisé en Trois chapitres :

- Chapitre 1 : Présentation de lieu de stage et Problématique
- Chapitre 2 : La surveillance médicale à distance à base d'internet des objets médicaux
- Chapitre 3 : Méthodes de réalisation et Implémentation de prototype

# Chapitre 1

## Présentation du lieu de stage et Problématique

### Introduction

L'objectif de ce chapitre est de faire une présentation générale de l'organisme d'accueil en précisant son historique, son activité, son organigramme, ses services, ainsi que le service biomédical dans lequel ce stage a été mise en œuvre, ainsi une présentation de la problématique constatée au sein du service.



FIGURE 1.1 – Accueil Ibn Rochd

### 1.1 Présentation du Centre Hospitalier Universitaire Ibn Rochd

Le Centre hospitalier universitaire Ibn Rochd (CHUIR / CHU Ibn Rochd) est parmi les plus importants centres universitaire hospitaliers du Maroc, situé dans la ville de Casablanca, affilié au ministère de la Santé. Ce centre hospitalier universitaire comprend quatre établissements médicaux : l'hôpital Ibn Rochd, l'hôpital du 20 août 1998, l'hôpital mère et père enfant adulte Abdel Rahim Al Harouchi et le centre d'examen et de traitement dentaire. Le CHU Hopital Ibn Rochd s'étend sur

une superficie de 45 hectares et est situé dans le centre-ville, à proximité d'autres établissements de santé publics, et de divers transports en commun à côté sud, ce qui le rend facile d'accès libre, Le CHU Ibn Rochd est connu à l'échelle nationale pour avoir des médecins spécialisés dans certaines maladies difficiles et complexes, telles que l'hématologie, l'oncologie, la chirurgie cardiovasculaire, la neurochirurgie, le traitement des brûlures, la chirurgie plastique, la chirurgie maxillo-faciale, la chirurgie thoracique et la chirurgie cardiovasculaire.



FIGURE 1.2 – Etablissements médicaux Ibno Rochd

### 1.1.1 Historique

Le CHUIR a été créé à partir des hôpitaux Maurice Gaud, Jules Colombani et l'hôpital militaire Jean Vial qui a été construit au cours des années trente.

En 1956, l'hôpital militaire prend le nom de l'Hôpital 20 Août 1953, les Hôpitaux Maurice Gaud et Jules Colombani ont été fusionnés sous le nom de l'Hôpital Ibn Rochd.

Par ailleurs, les locaux de l'Hôpital d'enfants, de même que le bâtiment des laboratoires d'analyses médicales biologiques et celui de l'anatomie pathologique furent construits en 1979.

Quant au Centre de Consultations et de Traitements Dentaires, il a été rattaché au CHUIR en 1981, année de l'ouverture de la Faculté de Médecine Dentaire de Casablanca.

Depuis 1983, le Centre Hospitalier Universitaire Ibn Rochd est érigé en établissement public, à caractère administratif, doté de la personnalité morale et de l'autonomie financière.

Date de création :	1921
Capacité litière fonctionnelle :	1020
Type d'architecture :	Pavillonnaire
Populations desservies :	Casablanca et le Sud Du Maroc
Adresse :	1, Quartier des Hôpitaux Casa
Standard téléphonique :	05 22 41 09 (LG)
Téléphones des urgences :	05 22 47 00 15/16/17
Fax :	05 22 29 94 83

TABLE 1.1 – Fiche Signalétique.

### 1.1.2 Organigramme de CHUIR

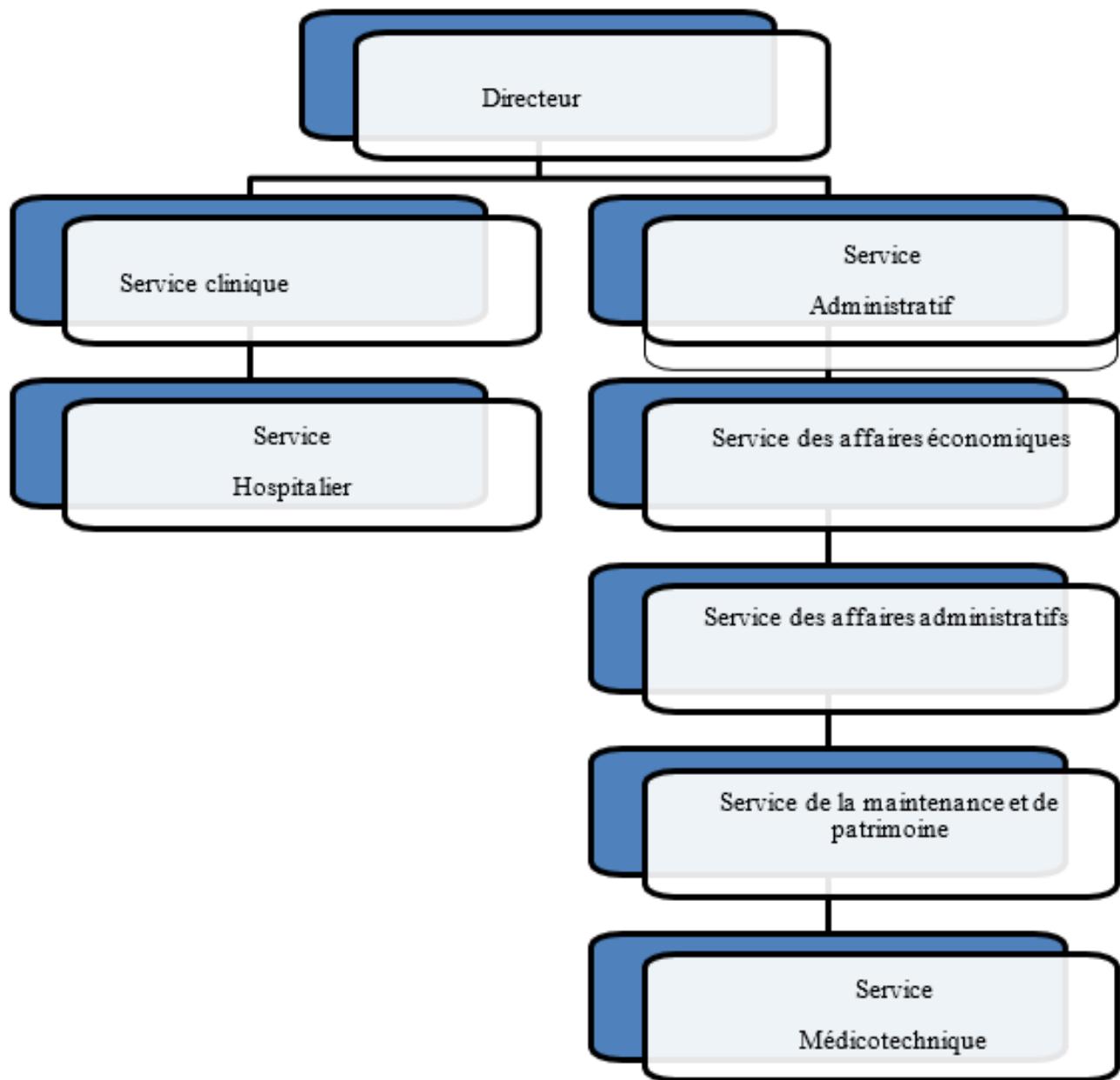


FIGURE 1.3 – Organigramme de l'hôpital

### 1.1.3 La présentation du service biomédical

La cellule biomédicale du Centre Hospitalier Universitaire Ibn Rochd sise à Casablanca a été créée en 1988, joue le rôle d'interface entre tous les services de l'établissement hospitalier et avec des interlocuteurs extérieurs, principalement avec les sociétés privées spécialisées dans la fourniture et le maintien des dispositifs médicaux.

## **Composition**

Le service est composé de :

- Chef de service du patrimoine et de la maintenance
- Chef du service biomédical
- Techniciens biomédicaux
- Secrétaires

## **Missions**

Le service biomédical a pour missions de :

- o Maintenir la performance des équipements dans le temps, adapter leur environnement pour répondre aux besoins exprimés ou implicites des utilisateurs.
- o Assurer la mise en conformité des équipements pour répondre aux textes de loi et réglementation en vigueur, dans le respect de l'environnement.
- o Assurer la sécurité des équipements, et les contrôles obligatoires.
- o Participer à l'investissement et l'achat des équipements biomédicaux, le service lance les consultations et l'ingénieur se charge de la partie administrative, les choix sont effectués en collaboration avec les services utilisateurs et avec la pharmacie pour les appareils nécessitant du consommable.
- o La planification des maintenances préventives.

## **1.2 Les tâches effectuées durant le stage**

Service / date	Désignation	Travail effectué
<b>Réanimation Chirurgicale des Urgences 07/04/2023</b>	TDM GE Optima	Maintenance corrective faite par la société T2S (Changement du bloc d'alimentation).
<b>Service de pneumologie 09/04/2023</b>	Respirateur GE	Maintenance préventive faite par la société T2S.
<b>Service d'oncologie 15/04/2023</b>	Table opératoire	Changement d'huile fait par les techniciens du service biomédical.
<b>Médecine interne 17/04/2023</b>	Aspirateur	Maintenance corrective faite par les techniciens du service biomédical.
<b>Réanimation Chirurgicale 19/04/2023</b>	Respirateur de réanimation GE	Maintenance corrective faite par la société T2S (Changement du contrôle valve d'air).
<b>Réanimation Chirurgicale 19/04/2023</b>	TDM GE Optima	La maintenance préventive faite par la société T2S (Nettoyage / Changement des filtres).

<b>Service d'oncologie 05/05/2023</b>	ECG EDAN	Maintenance corrective faite par les techniciens du service biomédical.
<b>Réanimation Chirurgicale des Urgences 09/05/2023</b>	Radiographie standard ITALRAY	Mise en marche par les techniciens du service biomédical.
<b>Réanimation Chirurgicale 17/05/2023</b>	Chauffe sérum	Maintenance corrective faite par les techniciens du service biomédical).

TABLE 1.2: Les tâches effectuées durant le stage

### 1.3 Problématiques observées



FIGURE 1.4 – Salle de réanimation

Au cours de notre stage, nous avons identifié une problématique majeure dans les salles de réanimation de l'hôpital CHUIR. Nous avons constaté que malgré le grand nombre de patients nécessitant un suivi continu de leur état de santé, les ressources médicales, notamment les lits médicaux et les moniteurs, sont limitées. Cette situation crée des défis importants en termes de surveillance et de prise en charge efficace des patients.

En raison du nombre restreint de lits médicaux disponibles, certains patients doivent attendre avant d'être admis en réanimation, ce qui peut compromettre leur état de santé et entraîner des retards dans les soins. De plus, les moniteurs, qui jouent un rôle essentiel dans la surveillance des signes vitaux tels que la fréquence cardiaque, la saturation en oxygène et la pression artérielle, sont également en nombre limité. Cela signifie que les patients qui ne sont pas directement surveillés peuvent présenter des risques élevés et ne pas recevoir une attention immédiate en cas de complications.

Face à cette problématique, notre objectif est de développer une solution innovante et efficace qui permettra de suivre à distance l'état des patients, en utilisant des technologies telles que IoT et les

dispositifs portables. Nous cherchons à concevoir un système embarqué qui permettra aux professionnels de la santé de surveiller en temps réel les signes vitaux des patients depuis une plateforme centralisée, à l'aide d'une application desktop intuitive et conviviale. Cette approche permettra de maximiser l'utilisation des ressources disponibles et d'améliorer la réactivité dans la prise en charge des patients, en réduisant les retards et en prévenant les complications potentielles.

La résolution de cette problématique présente un intérêt crucial pour l'amélioration de la qualité des soins dans les services de l'hôpital. En fournissant un suivi à distance fiable et en temps réel, notre solution contribuera à optimiser les ressources médicales, à réduire les risques pour les patients et à faciliter la prise de décisions médicales éclairées.

## Conclusion

Dans ce premier chapitre, nous avons présenté brièvement le CHUIR, le service biomédical et ces missions et les tâches affectueuses durant notre stage, enfin une présentation sur la problématique observée. Le chapitre suivant sera consacré au l'état de l'art sur la surveillance médicale à distance à base d'internet des objets médicaux et les services et architectures web.

# Chapitre 2

# La surveillance médicale à distance à base d'internet des objets médicaux

## Introduction

L'évolution des technologies de l'information et de la communication a ouvert la voie à de nouvelles avancées dans le domaine de la santé. L'un des développements majeurs de ces dernières années est l'émergence de l'Internet des objets (IdO) et son application dans le domaine médical

L'Internet des objets (IdO) et son application dans le domaine médical L'IdO médical, également connu sous le nom d'Internet des objets médicaux (IdOM), révolutionne la manière dont la surveillance médicale est effectuée en permettant une surveillance à distance des patients.

L'objectif de ce chapitre est de présenter un aperçu général sur cette nouvelle méthode. Nous allons aborder le concept de l'IoT , ses différents processus , ses avantages et ses inconvénients . Puis, nous explorerons le concept de la surveillance médicale et ses types. Ainsi que les différents aspects de l'Internet des objets médicaux, ses avantages et ses limites, aussi que les services et les architectures de réseau IoTM.

## 2.1 L'internet des objets



FIGURE 2.1 – Internet des objets

## 2.1.1 Définition

Le terme « internet of things » en français internet des objets est né en 1999 au centre, MTT (Massachusetts Institute of Technology), fondé par Kevin Ashton, un chercheur britannique, pionnier dans son domaine.

Internet des objets (IoT) désigne l'interconnexion de millions d'appareils et de capteurs intelligents connectés à Internet. Ces derniers collectent et partagent des données qui seront utilisées et analysées par plusieurs organismes, dont des entreprises, des villes, des gouvernements, des hôpitaux et des particuliers. L'avènement de l'IoT a été possible grâce à l'apparition des nouvelles technologies électroniques telles que les capteurs, les actionneurs, diverses cartes DIY (Do It Yourself) et des réseaux sans fil.[Np, 2020] Des objets jusqu'à présent inanimés peuvent désormais être équipés d'un capteur intelligent ou plusieurs qui collectent des données et les transferts à un réseau.

## 2.1.2 Les différents processus de l'IoT

- **Capter** : désigne l'action de transformer une grandeur physique en un signal numérique.
- **Concentrer** : permet d'interfacer un réseau spécialisé d'objets à un réseau IP standard (exemple : wifi).
- **Stocker** : qualifie le fait d'agrégier des données brutes, produites en temps réel, métatagées, arrivant de façon non prédictible.
- **Présenter** : indique la capacité de restituer les informations de façon compréhensible par l'homme, tout en lui offrant un moyen d'agir et/ou d'interagir.

## 2.1.3 Les objets Connectés

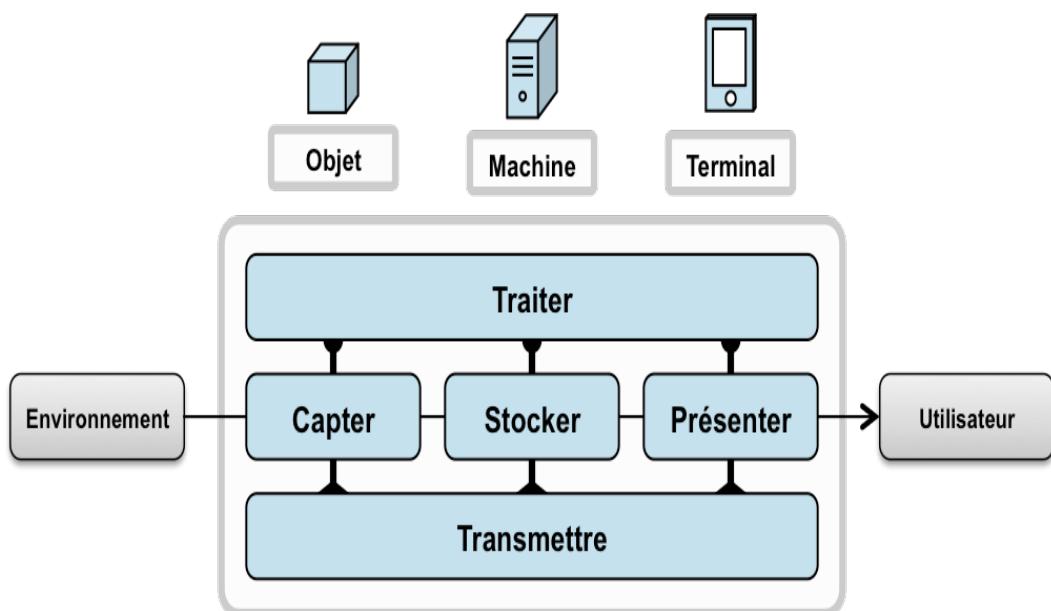


FIGURE 2.2 – Architecture d'un objet connecté

## Définition

Un objet connecté est un appareil composé de capteurs qui envoient des données vers une application mobile ou un service web pour de multiples domaines d'application. Les objets connectés s'appuient sur les innovations technologiques majeures telles que le Cloud Computing, et l'avancée en général des techniques informatiques. De plus en plus performants, ils peuvent être reliés à des services pour mesurer, partager, suivre, analyser les données quotidiennes de l'individu et lui donner des conseils. Depuis peu, les objets connectés commencent à prendre part dans notre vie quotidienne à l'image de : Ordinateurs, téléviseurs, cameras IP, les smartphones, les montres connectées et bien d'autres.

### Les différents éléments d'un objet connecté

**Les Capteurs** Les capteurs sont des éléments qui transforment une grandeur physique en information électrique. Ils rendent compte de l'état du système à un moment donné. L'information est délivrée sous la forme d'un signal qui sera exploité par une interface programmable.

**Les Actionneurs** Partie d'une machine ou d'un système de commande à distance qui permet de convertir l'énergie reçue en travail utile pour exécuter les tâches d'un système automatisé.

**Le microcontrôleur** Un microcontrôleur se présente sous la forme d'un circuit intégré qui contient en interne, l'équivalent de la structure complète d'un microordinateur. C'est à dire qu'il rassemble dans un même boîtier un microprocesseur, plusieurs types mémoires et des périphériques de communication (Entrées-Sorties).

**Les bus** Un bus est un ensemble de lignes, transportant des informations codées binaires. Chacune de ces lignes est affectée d'un poids binaire. C'est par l'intermédiaire de ces lignes que s'effectuent les échanges entre les différents éléments du système. On distingue 3 types de bus : le bus de données, le bus d'adresse et le bus de contrôle.

**Les périphériques** Sont des circuits électroniques intégrés au microcontrôleur capable d'effectuer des tâches spécifiques. On peut mentionner entre autres :

- Les convertisseurs analogiques-numériques(CAN) : donnent un nombre binaire à partir d'une tension électrique.
- Les convertisseurs numériques-analogiques (CNA) : donnent une tension électrique à partir d'un nombre
- Les générateurs de signaux à modulation de largeur d'impulsion (MLI, ou en anglais,PWM pour pulse Width Modulation).
- Les timers : Compteurs d'impulsions d'horloge interne ou d'événements externes
- Les chiens de garde (Watch Dog)
- Les comparateurs : Comparant deux tensions électriques
- Les contrôleurs de bus de communication (UART, IIC, SSP, CAN, flexray, USB, Ethernet, etc.)

- Les entrées/sorties : Permettent au microcontrôleur de communiquer avec le monde extérieur. C'est donc là qu'elles vont être connectées aux claviers, aux afficheurs, aux pousoirs, aux moteurs et relais, etc.

#### 2.1.4 Les avantages de l'IoT

Les principaux bénéfices que peuvent avoir un système doté d'un système IOT sont :

- **Amélioration de la productivité** : l'IoT permet la surveillance, le monitoring et le contrôle de différentes tâches, ce qui optimise les différentes opérations qui augmentent la productivité et l'efficacité
- **Analyses prédictives** : grâce à la collecte de nombreuses données, les nouvelles technologies de l'IoT permettent d'examiner les patrons récurrents et contribuent à l'analyse prédictive qui peut être principalement utilisée en maintenance. Ces informations précises vont servir à améliorer les tâches et les services existants
- **Rapidité d'action** : les données permettent de suivre en temps réel et même à distance les systèmes mis en place. Elles facilitent l'optimisation des interventions de maintenance, mais aussi donnent un avantage stratégique à l'entreprise dans le suivi de l'évolution des marchés.
- **Diminution des erreurs humaines** : grâce à la complémentarité des technologies comme l'intelligence artificielle, l'IoT permet de minorer les erreurs humaines dues à des tâches répétitives.

#### 2.1.5 Les inconvénients de l'IoT

Les enjeux des cyberattaques sont élevés pour les consommateurs lorsqu'il est question d'IoT, et il est essentiel de comprendre et de contrôler les nouveaux types de conséquences. Citons par exemple :

- Une attaque contre un appareil médical peut induire des dommages physiques, voir la perte d'une vie.
- Le brouillage d'un capteur utilisé pour des compteurs intelligents pourrait provoquer une panne de courant et générer divers inconvénients causés par le manque d'électricité
- Le piratage d'une voiture sans chauffeur risquerait de se traduire par une perte de contrôle du véhicule sur une autoroute bondée.
- L'exploitation des faiblesses d'un appareil IoT pourrait permettre à un individu malveillant d'accéder à des données hautement sensibles telles que des renseignements bancaires, personnels ou de santé.
- Un piratage de système dans une smart home peut engendrer un disfonctionnement total à l'intérieur de la maison (ouverture des portes, lancer les matériels de la maison sans contrôle)

#### 2.1.6 La sécurité dans l'IoT

La sécurité est l'élément le plus important qui pose des obstacles devant l'adaptation des êtres humains des applications de l'IoT suite aux menaces concernant la vie privée des utilisateurs. En effet ,sécuriser un système informatique implique directement la garantie des objectifs suivants :

- **L'Authentification :** En contrôlant et identifiant les nœuds au sein d'un réseau, l'authentification permet la coopération entre eux. Cette phase précède toutes les autres phases, et on ne peut jamais assurer une confidentialité et intégrité de donnée si l'authentification est mal gérée, et un simple attaquant peut rejoindre le réseau et injecte des messages erronés. L'utilisation d'un code d'authentification de message MAC3 permet d'assurer l'authentification de l'origine et l'intégrité de message.
- **L'intégrité :** Les données reçues par le nœud récepteur doivent être identiques à celles envoyées par le nœud émetteur sans aucune altération. Pour chaque message envoyé, une empreinte digitale est générée par une fonction de hachage pour assurer l'intégrité.
- **La confidentialité :** Cette propriété est assurée par l'utilisation de clés cryptographiques (symétrique ou asymétrique). Elle consiste à préserver le secret du message échangé et de ne pas le révéler aux adversaires.
- **La disponibilité :** Le réseau doit être disponible à tout moment et autorise les parties communicantes à utiliser du medium quand c'est nécessaire.
- **La fraîcheur :** Les données échangées doivent être actuelles, et ne sont pas des réinjections des échanges précédents interceptés par des attaquants.

Les propriétés citées précédemment constituent un défi majeur devant l'Internet des Objets avant que ce paradigme devient une réalité. La vulnérabilité des objets de l'IoT aux attaques de sécurité est due à l'impossibilité d'appliquer des standards de sécurité classiques à cause de la densité importante du réseau et sa topologie dynamique, la limite des nœuds en ressource (stockage, calcul, énergie) et le type de communication (sans-fil).

## 2.2 L'internet des objets médicaux

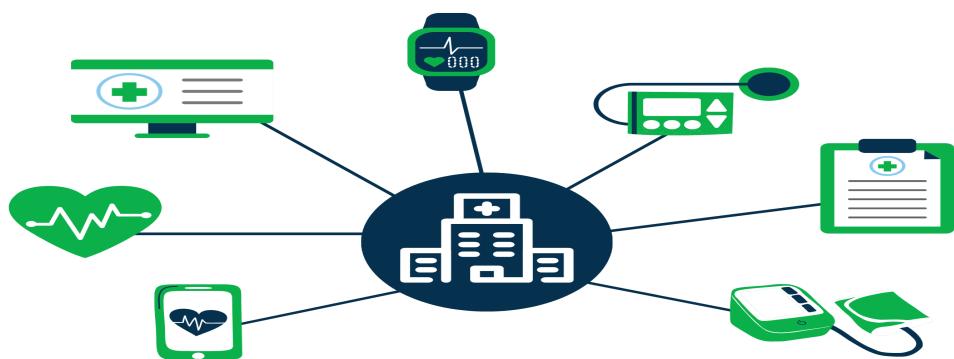


FIGURE 2.3 – Internet des objets médicaux

### 2.2.1 La surveillance médicale

[Çetin, ] La surveillance médicale est un processus par lequel les professionnels de la santé suivent régulièrement l'état de santé d'une personne dans le but de détecter et de prévenir les maladies, d'évaluer l'efficacité des traitements ou des interventions médicales, et de promouvoir le bien-être

général. Elle peut être effectuée de différentes manières, notamment par le biais d'examens médicaux réguliers, de tests de dépistage, de suivi des signes vitaux, de surveillance des symptômes, d'analyses de laboratoire et d'autres mesures diagnostiques.

La surveillance médicale peut être appliquée à divers contextes, tels que le suivi des patients atteints de maladies chroniques, le suivi des patients après des interventions chirurgicales, le suivi des femmes enceintes ou le suivi des travailleurs exposés à des risques professionnels. Elle permet aux professionnels de la santé d'identifier les problèmes de santé dès leur apparition ou même avant qu'ils ne se manifestent, ce qui permet une intervention précoce et une prise en charge adéquate.

## 2.2.2 Types de surveillance médicale

Dans l'espace IoT dédié à la santé, la Surveillance continue des patients est un élément clé pour réduire le risque de réadmission, gérer plus efficacement les maladies chroniques et améliorer les résultats des patients. La surveillance continue des patients peut être divisée en deux catégories :

- 1 **Surveillance des patients hospitalisés** : les équipes de soins utilisent des dispositifs médicaux portables et d'autres appareils pour surveiller les signes vitaux et les conditions médicales des patients, sans devoir envoyer un/e infirmier/ère pour surveiller un patient plusieurs fois par jour. Les équipes de soins peuvent recevoir des notifications lorsqu'un patient a besoin d'une attention critique et hiérarchiser efficacement son temps.
- 2 **Surveillance à distance des patients** : les équipes de soins utilisent des dispositifs portables médicaux pour surveiller les patients à l'extérieur de l'hôpital afin de réduire le risque de réadmission. Les données recueillies auprès des patients atteints de maladies chroniques et des patients en rééducation peuvent permettre de s'assurer que les patients respectent leur traitement et que les alertes de détérioration des patients sont transmises aux équipes de soins avant que cet état ne devienne critique.

## 2.2.3 Définition de l'IdOM

L'Internet des objets médicaux (IdOM) est le point d'intersection entre les dispositifs médicaux et l'Internet des objets (IoT). Elle est l'avenir des systèmes de santé actuels où chaque appareil médical sera connecté et surveillé sur Internet via des professionnels de la santé.

L'IdOM joue un rôle central dans l'amélioration de l'efficacité, de la réactivité et la surveillance à distance des dispositifs médicaux.

L'IdOM est traitée en connectant et en communiquant en M2M des dispositifs médicaux équipés de Wi-Fi. Les données reçues sont stockées dans la base de données du serveur Cloud puis analysée.

## 2.2.4 Avantages et inconvénients d'utilisation de technologie IdOM

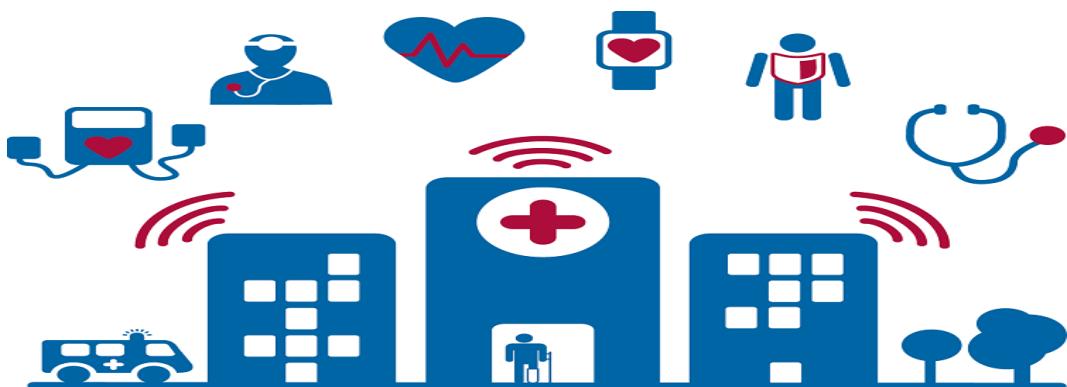


FIGURE 2.4 – Internet des objets médicaux options

Les avantages d'utilisation de technologie IdOM sont nombreux, nous pouvons citer :

### Avantages pour les patients :

- Interventions en temps réel dans les situations d'urgence.
- Réduction des coûts.
- Réduction de la morbidité et de la charge financière en raison de la diminution des visites de suivi.

### Avantages pour les prestataires de services de soins de santé :

- Interventions en temps réel dans les situations d'urgence
- Réduction des coûts.
- Réduction de la morbidité et de la charge financière en raison de la diminution des visites de suivi.
- Utilisation optimale des ressources et des infrastructures
- Réduction du temps de réponse en cas d'urgence médicale

### Parmi les inconvénients, nous pouvons citer :

- La protection des données, la vie privée et la sécurité sont souvent les principales inquiétudes des sceptiques de l'IdO. Pour calmer ces inquiétudes, il serait utile de donner aux clients les informations relatives au lieu de stockage et à la nature des données qui les concernent.
- L'installation des objets connectés est couteuse.

## 2.2.5 Rôle de l'IdOM dans le domaine de la santé

Les défis quotidiens à venir dans le domaine de la santé ont créé de nouvelles exigences pour les prestataires de soins de santé afin de fournir avec succès des services de santé de haute qualité. L'IdO

avec ses avantages tels que la surveillance en temps réel, la fourniture d'une solution plus mature et efficace pour collecter les données des patients et le suivi des activités des patients et du personnel répond aux nouvelles exigences telles que :

- Système d'information clinique pour fournir une aide à la décision.
- Système de santé avancé axé sur une approche de traitement moderne et la conception d'un système de prestation de soins de santé.
- Un programme de soins continus bien structuré conçu par une équipe multidisciplinaire.
- Obtenez une meilleure compréhension de l'état de santé d'un patient dans une série de "phases" et de besoins de soins complexes.

L'IoT est utile dans les soins de santé à différentes phases de l'interaction entre le patient et le système de santé. Cela commence lorsque le patient appelle l'hôpital pour prendre des rendez-vous ou se rend directement à l'hôpital en cas d'urgence. Dès lors, grâce à la surveillance en temps réel de l'état d'un patient à des dispositifs médicaux intelligents connectés à une application pour smartphone, les prestataires de soins de santé peuvent obtenir les données de santé nécessaires qu'un médecin peut utiliser pour analyser l'état du patient et planifier un meilleur traitement.

## 2.3 Architecture et Protocoles dans IdO et IdOM

### 2.3.1 Architecture de réseau IdOM

L'architecture proposée repose sur la communication entre l'unité de soins dans le domicile du patient et le terminal de soins de santé chez le médecin :



FIGURE 2.5 – Architecture de réseau IdOM

- 1 La première couche permet de détecter et collecter des données à partir de capteurs biométriques portés sur soi. Des unités de calcul et de traitement intelligentes connectent les dispositifs matériels et la surface de peau du patient.
- 2 Les données collectées sont traitées et transférées à la couche de service, qui se compose de dispositifs de stockage et d'organisation de données et de protocoles filaires/sans fil tels que Bluetooth, Zigbee, RFID, Wi-Fi, Ethernet et les réseaux 3G/4G. Ceux-ci sont reliés à la couche IdOM qui communique les paramètres mesurés aux établissements tels que les hôpitaux, les centres d'urgence et les ambulances. L'examen de santé global d'un groupe de patients est effectué par les médecins à l'aide de dispositifs matériels et logiciels efficaces, qui analysent automatiquement les variations des paramètres de chaque patient et identifient leur état physique sur une période donnée. Cette couche fournit des services de Cloud Computing, de protection des données et de confidentialité des patients.
- 3 La troisième couche d'application utilise l'interface entre les médecins et les patients à distance pour fournir [Cetin, ] facilement des informations sur l'état de santé à la demande ou sur une base régulière.

### 2.3.2 Les protocoles de communication :

[Ap, 2018] Les protocoles de communication permettent d'effectuer l'échange d'information sur un moyen de communication physique :

- **HTTP** : HTTP est un protocole omniprésent dans l'Internet, qui permet d'échanger des données, par exemple lors de la consultation de sites Web. Il est cependant :

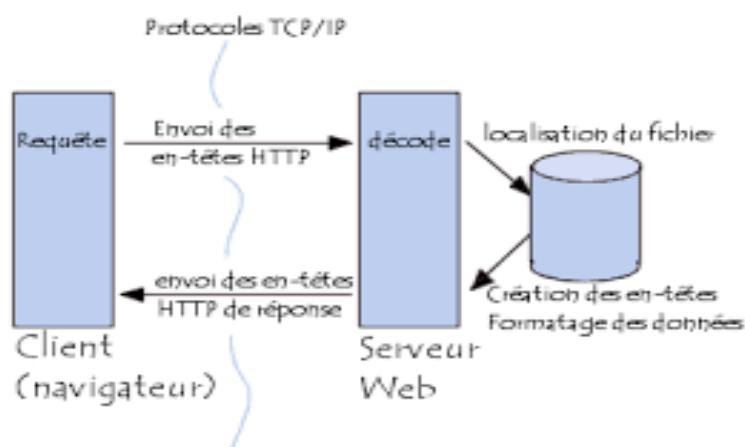


FIGURE 2.6 – protocole HTTP

- o Assez verbaux.
- o Plus adapté au API haut niveau.
- o Communication entre applications
- o Plus lourds (données).
- o Sécurité : HTTPS pour version sécurisée
- **MQTT** : est un protocole alternatif très utilisé dans l'Internet des Objets. En effet, MQTT (pour Message Queuing Telemetry Transport) est un protocole adapté aux connexions qui intègrent une partie mobilité entre les clients et le serveur qui va stocker les données.

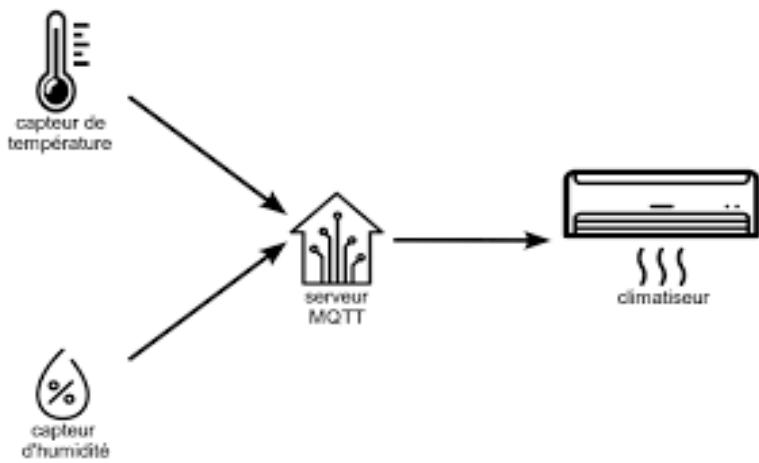


FIGURE 2.7 – protocole MQTT

- **COAP :** (Constrained Application Protocol) : est un protocole de transport de données utilisé dans l'Internet des Objets. Ce protocole s'adresse davantage à des environnements contraints qui possèdent peu de bande passante pour échanger des données. En effet, le nombre d'octets est limité par envoi et la surcharge du protocole est très faible. C'est donc un protocole très utile pour les environnements contraints avec des équipements à faible capacité de traitement et de transfert.

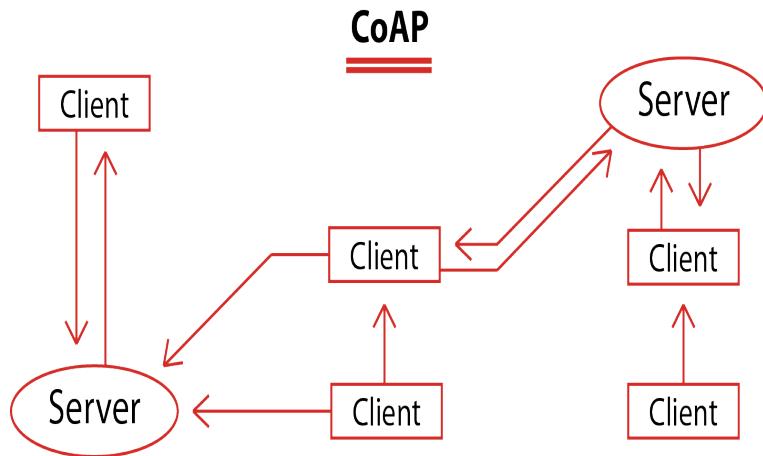


FIGURE 2.8 – COAP Diagramma

## Conclusion

Le cadre de notre Stage était principalement sur le développement d'une application IoT connecter avec moniteur portable à base de carte ESP8266 pour le suivi de l'état des patients à distance, dans le chapitre suivant nous serons présentons les méthodes et les techniques utilisés pour la réalisation de ce travail ainsi que les résultats obtenus.

# Chapitre 3

## Méthodes de réalisation et Implémentation du prototype

### Introduction

Dans ce chapitre, Nous allons définir les méthodes utiliser et plan d'action mise en compte pour la réalisation de notre projet durant notre stage, ainsi que les résultats obtenu et Apport du Stage.

#### 3.1 Planification du projet

La planification est parmi les phases d'avant-projet les plus importantes. Le diagramme de GANTT est l'un des outils de planification de projet qui permet de répertorier toutes les tâches à accomplir pour mener le projet à bien, et indique la date à laquelle ces tâches doivent être effectuées.

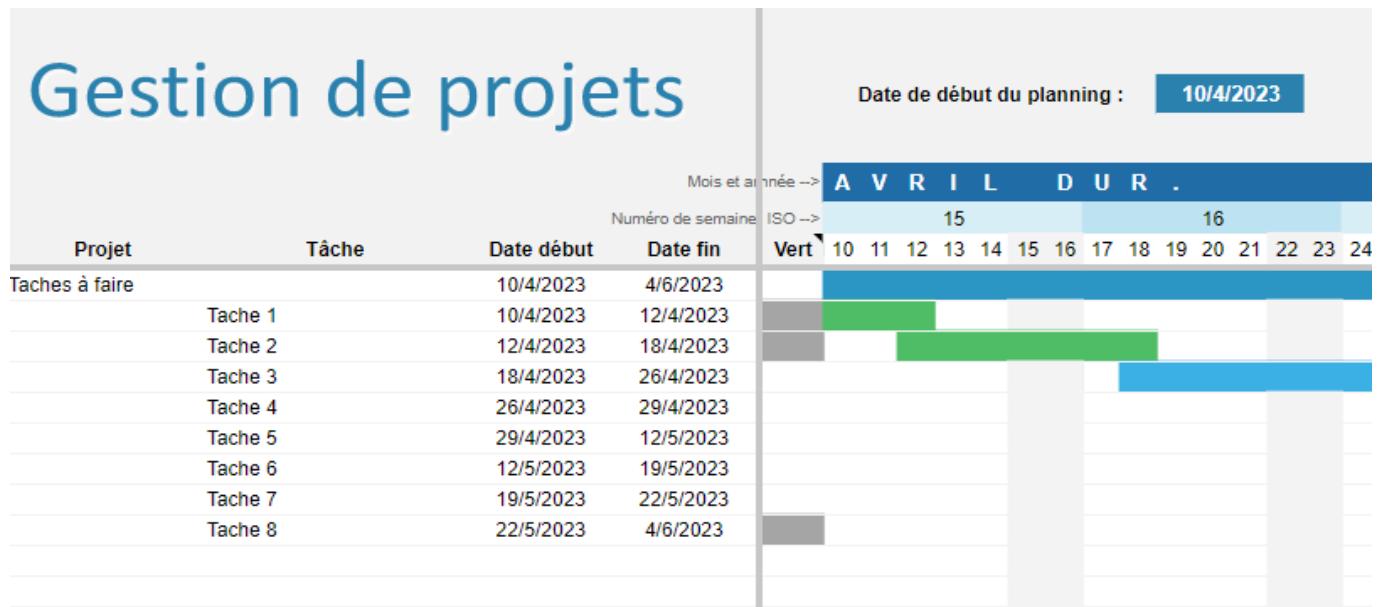


FIGURE 3.1 – Diagramme de GANTT

- **Tache 1 :** Recherche et sélection des capteurs appropriés pour collecter les signes vitaux des patients tels que l'ECG, le SpO2 et la pression sanguine.

- **Tache 2** : Assemblage du circuit électronique avec le microcontrôleur ESP8266 pour connecter les capteurs et collecter les données.
- **Tache 3** : Programmation du microcontrôleur pour configurer la connexion Wi-Fi et la transmission des données via le protocole MQTT.
- **Tache 4** : Création d'une base de données pour stocker les données collectées des capteurs
- **Tache 5** : Développement d'une application Qt pour créer une interface utilisateur pour les professionnels de la santé pour accéder aux données collectées des capteurs.
- **Tache 6** : Configuration du back-end Python pour recevoir les données du microcontrôleur ESP8266 et les stocker dans la base de données
- **Tache 7** : Intégration de l'application Qt avec la base de données pour afficher les données collectées en temps réel pour les professionnels de la santé.
- **Tache 8** : Tests et débogage de l'ensemble du système pour s'assurer que les données collectées des capteurs sont correctement affichées dans l'interface utilisateur et stockées dans la base de données. Réalisation de toutes les documentations nécessaires pour le projet.

## 3.2 Méthodes de réalisation matérielle

### 3.2.1 Les Capteurs

Comme nous l'avons mentionné dans le plan d'action, nous devons d'abord sélectionner les capteurs appropriés pour collecter les signes vitaux, pour cela nous avons choisi des capteurs qui sont disponibles sur le marché, mais cela ne dit pas que seuls ces paramètres que nous pouvons capturer, notre solution est capable de transmettre n'importe quel paramètre que nous voulons, il suffit de le connecter pour le surveiller et de le définir dans l'application, pour notre prototype nous avons utilisé :

- **Capteur de température analogique LM35 :**

Le LM35 est un capteur de température centigrade de précision peu coûteux fabriqué par Texas Instruments. Il fournit une tension de sortie linéairement proportionnelle à la température centigrade.

Le capteur LM35 ne nécessite aucun étalonnage ou réglage externe pour fournir des précisions de  $\pm 0,5^\circ\text{C}$  à température ambiante et de  $\pm 1^\circ\text{C}$  sur la plage de températures de  $-50^\circ\text{C}$  à  $+155^\circ\text{C}$



FIGURE 3.2 – Capteur de température analogique LM35

### — Module de fréquence cardiaque oxymètre MAX30100 :

Le clic de fréquence cardiaque transporte l'oxymétrie de pouls et le capteur de fréquence cardiaque MAX30100 intégrés

Le capteur MAX30100 est utilisé à la fois comme moniteur de fréquence cardiaque et comme oxymètre de pouls. Ces fonctionnalités sont rendues possibles par la construction de ce capteur qui se compose de deux LED, d'un photodétecteur, d'une optique optimisée et de composants de traitement du signal à faible bruit.[mic, ]

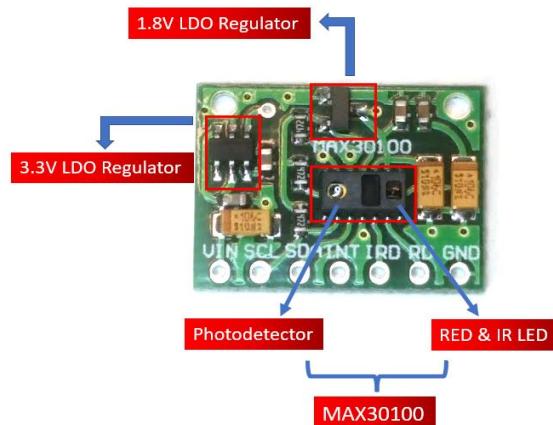


FIGURE 3.3 – Oxymètre MAX30100

### Fonctionnement du capteur d'oxymètre de pouls MAX30100

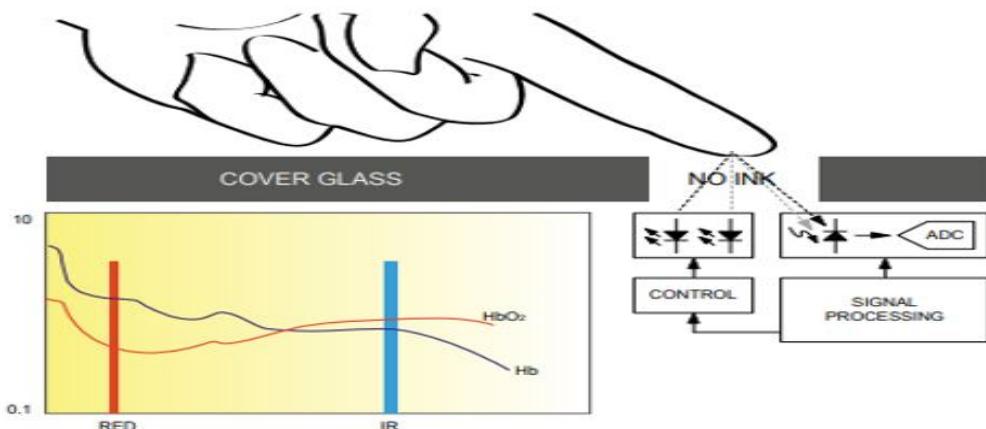


FIGURE 3.4 – Schéma fonctionnel du système MAX30100

**Oxymètre de pouls :** Pour trouver la concentration en oxygène dans le sang (%), il faut d'abord savoir qu'à l'intérieur de notre sang l'hémoglobine est responsable du transport de l'oxygène. Lorsqu'une personne tient un oxymètre de pouls, la lumière de l'appareil traverse le sang dans les doigts. Ceci est utilisé pour détecter la quantité d'oxygène en mesurant les changements d'absorption de la lumière dans le sang oxygéné et désoxygéné.

Le capteur MAX30100 se compose de deux LED (rouge et IR) et d'une photodiode. Ces deux LED sont utilisées pour la mesure SPO<sub>2</sub>. Ces deux LED émettent des lumières à différentes longueurs d'onde, 640nm pour la LED rouge et 940nm pour la LED IR. A ces longueurs d'onde particulières, l'hémoglobine oxygénée et désoxygénée ont des propriétés d'absorption très différentes.

La Figure 3.4 est tiré de la fiche technique du MAX30100 IC. Vous remarquerez peut-être la différence indiquée dans le graphique entre l'HbO<sub>2</sub> qui est de l'hémoglobine oxygénée et l'Hb qui est de l'hémoglobine désoxygénée à deux longueurs d'onde différentes.

L'hémoglobine oxygénée absorbe plus de lumière infrarouge et réfléchit la lumière rouge tandis que l'hémoglobine désoxygénée absorbe plus de lumière rouge et réfléchit la lumière infrarouge. La lumière réfléchie est mesurée par le photodétecteur. Le capteur MAX30100 lit ces différents niveaux d'absorption pour trouver la concentration en oxygène dans le sang (SPO<sub>2</sub>). Le rapport de la lumière IR et ROUGE reçue par le photodétecteur nous donne la concentration en oxygène du sang.

**Mesure de la fréquence cardiaque :** Pour mesurer la fréquence cardiaque, nous n'avons pas besoin de la LED rouge, seule la LED IR est nécessaire. En effet, l'hémoglobine oxygénée absorbe plus de lumière infrarouge.

Le rythme cardiaque est le rapport de temps entre deux battements cardiaques consécutifs. De même, lorsque le sang humain circule dans le corps humain, ce sang est comprimé dans les tissus capillaires. En conséquence, le volume des tissus capillaires est augmenté mais ce volume est diminué après chaque battement cardiaque. Ce changement de volume des tissus capillaires affecte la lumière infrarouge du capteur, qui transmet la lumière après chaque battement cardiaque.

Le fonctionnement de ce capteur pourrait être vérifié en plaçant un doigt humain devant ce capteur. Lorsqu'un doigt est placé devant ce capteur de pouls, la réflexion de la lumière infrarouge est modifiée en fonction du volume de sang à l'intérieur des vaisseaux capillaires. Cela signifie que pendant le rythme cardiaque, le volume de sang dans les vaisseaux capillaires sera élevé, puis sera faible après chaque battement de cœur. Ainsi, en modifiant ce volume, la lumière LED est modifiée. Ce changement de la lumière LED mesure le rythme cardiaque d'un doigt. Ce phénomène est connu sous le nom de "photopléthysmogramme".

### 3.2.2 Les Microcontrôleurs

#### Microcontrôleur NodeMCU ESP8266

Pour le microcontrôleur, nous avons choisi le module nodemcu esp8266 pour diverses raisons, notamment le fait qu'il dispose du module wifi et peut facilement interagir avec le broker mqtt tout en disposant également du GPIO nécessaire et possède une capacité de mémoire et de calcul supérieure aux Arduino. Idéal pour notre projet connecté.



FIGURE 3.5 – NodeMCU ESP8266

**Caractéristiques du microcontrôleur :** Le microcontrôleur NodeMCU ESP8266 utilise le microprocesseur Tensilica 32-bit RISC CPU Xtensa LX106. Ce processeur fonctionne à une fréquence d'horloge de 80 MHz. Il possède une mémoire RAM de 64 kB, EEPROM de NC kB et aussi une mémoire Flash de 4000 kB (pour la programmation et l'enregistrement de données).

- CPU Tensilica 32-bit RISC CPU Xtensa LX106
- Voltage : 3.3V ou 5V
- Flash : 4000 kB
- RAM : 64 kB
- EEPROM : NC kB
- Clock speed : 80MHz
- WiFi : Yes
- Bluetooth : No
- SD : No

#### Arduino Uno

Carte Arduino Uno basée sur un ATmega328 cadencé à 16 MHz. Des connecteurs situés sur les bords extérieurs du circuit imprimé permettent d'enficher une série de modules complémentaires. Cette carte peut se programmer en USB avec le logiciel Arduino IDE (cordon USB à prévoir). Le microcontrôleur ATMega328 contient un bootloader qui permet de modifier le programme sans passer par un programmateur.

Des connecteurs situés sur les bords extérieurs du circuit imprimé permettent d'enficher une série de modules complémentaires.

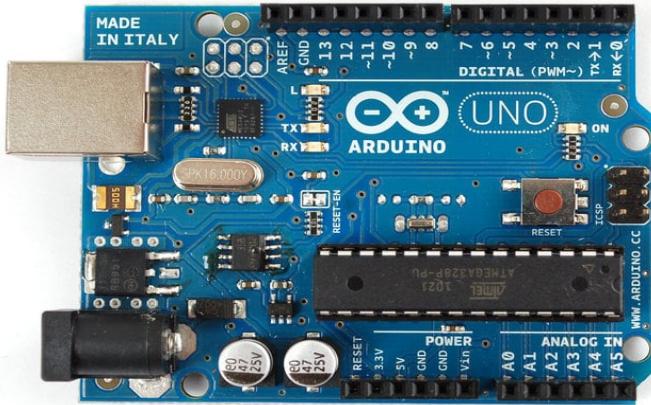


FIGURE 3.6 – Arduino Uno

#### Caractéristiques du microcontrôleur :

- Alimentation : - via port USB ou - 7 à 12 V sur connecteur alim 5,5 x 2,1 mm
- Microcontrôleur : ATMega328
- Mémoire flash : 32 kB
- Mémoire SRAM : 2 kB
- Mémoire EEPROM : 1 kB
- Interfaces : - 14 broches d'E/S dont 6 PWM - 6 entrées analogiques 10 bits - Bus série, I2C et SPI
- Intensité par E/S : 40 mA
- Cadencement : 16 MHz
- Gestion des interruptions

### 3.2.3 Interface de sortie

#### LCD I2C

I2C\_LCD est un module d'affichage facile à utiliser, il peut faciliter l'affichage. Son utilisation peut réduire la difficulté de fabrication, afin que les créateurs puissent se concentrer sur le cœur du travail.

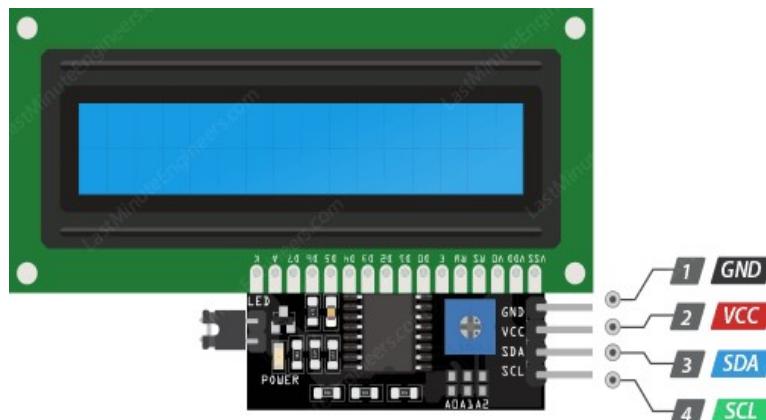


FIGURE 3.7 – LCD I2C

#### Description :

- Seules 2 broches Arduino sont occupées (Utilisez l'interface I2C).
- Prend en charge le mode I2C standard (100Kbit/s) et le mode I2C rapide (400Kbit/s).
- Compatible avec plusieurs niveaux logiques de communication : 2,8 à 5 VDC.
- Bibliothèque Arduino prise en charge, utilisez une ligne de code pour compléter l'affichage.
- Intégrer 7 tailles de polices ASCII, 5 fonctions graphiques.
- Fournir un logiciel de conversion de données d'image dédié (Bitmap Converter).
- La plupart des opérations complexes sont traitées par le contrôleur indépendant I2C LCD, ce qui permet d'économiser les ressources du contrôleur utilisateur.
- Prend en charge la fonction de curseur, peut configurer 16 fréquences de scintillement du curseur.
- Prend en charge le réglage de la luminosité du rétroéclairage à 128 niveaux.
- Prend en charge le réglage du contraste de l'écran à 64 niveaux.
- Prend en charge la modification de l'adresse de l'appareil.
- Prend en charge 127 travaux I2C LCD en parallèle.
- Lors du débogage du code, il peut remplacer le moniteur série pour surveiller l'état d'exécution du programme.
- Deux méthodes de récupération anormale sont fournies : réinitialiser et restaurer les paramètres d'usine.
- Compatible with Grove interface and 4Pin-100mil interface (under the Grove socket).
- 4 symmetrical fixed hole design for easy user installation.
- China style unique appearance.

## **BUZZER :**

Buzzer ce composant piézoélectrique qui transforme l'énergie électrique en vibration (son) à environ 400Hz. Le buzzer piézoélectrique est constitué d'un diaphragme piézoélectrique, accepte souvent une plage de tension importante : 3V à 20V par exemple. Et peut être équipé d'un oscillateur interne à fréquence fixe (Buzzer actif).

Par rapport aux transducteurs sonores électromagnétiques traditionnels, les transducteurs piézocéramiques ont une conception simple. L'émetteur piézocéramique se compose d'une plaque métallique sur laquelle est appliquée une céramique piézoélectrique avec une pulvérisation conductrice. La plaque et la pulvérisation sont les contacts de l'émetteur piézoélectrique (buzzer passif), et le dispositif a une polarité – plus et moins.

Le principe de fonctionnement des émetteurs est basé sur l'effet découvert par les frères Curie en 1880. Dans les piézocristaux, des charges électriques sont générées sous l'action de forces mécaniques de cisaillement, de flexion ou de torsion. Outre l'effet « direct », il existe également l'effet inverse : si l'on applique de l'électricité au cristal, celui-ci commence à se déformer. Les oscillations fréquentes du cristal créent une onde sonore d'une fréquence donnée.



FIGURE 3.8 – BUZZER

## **LEDs :**

Une LED (en français : DEL : diode électroluminescente) est un composant électronique et optique, qui en étant traversé par du courant électrique, émet une lumière d'une intensité diffuse. Les LED consomment peu d'électricité.

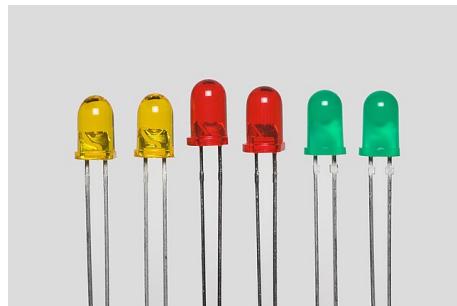


FIGURE 3.9 – LED's

### 3.3 Méthodes de réalisation Logicielle

Pour la réalisation de notre application Desktop qui va nous servir à transmettre et enregistrer les données propres de chaque patient, nous allons faire appel en premier lieu au divers langages et logiciels utilisés pour la concevoir.

#### 3.3.1 Python :

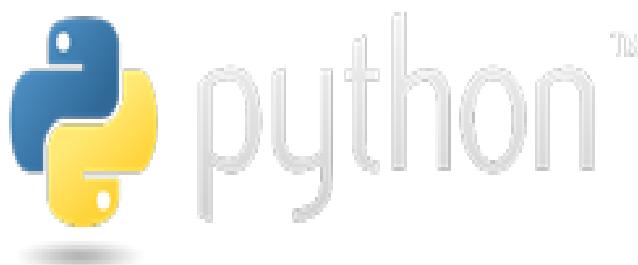


FIGURE 3.10 – Python

Python est un langage de programmation interprété multi-paradigme. Il favorise la programmation impérative structurée, et orientée objet. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions ; il est ainsi similaire à Perl, Ruby, Scheme, Smalltalk et Tcl.

#### 3.3.2 Qt designer :

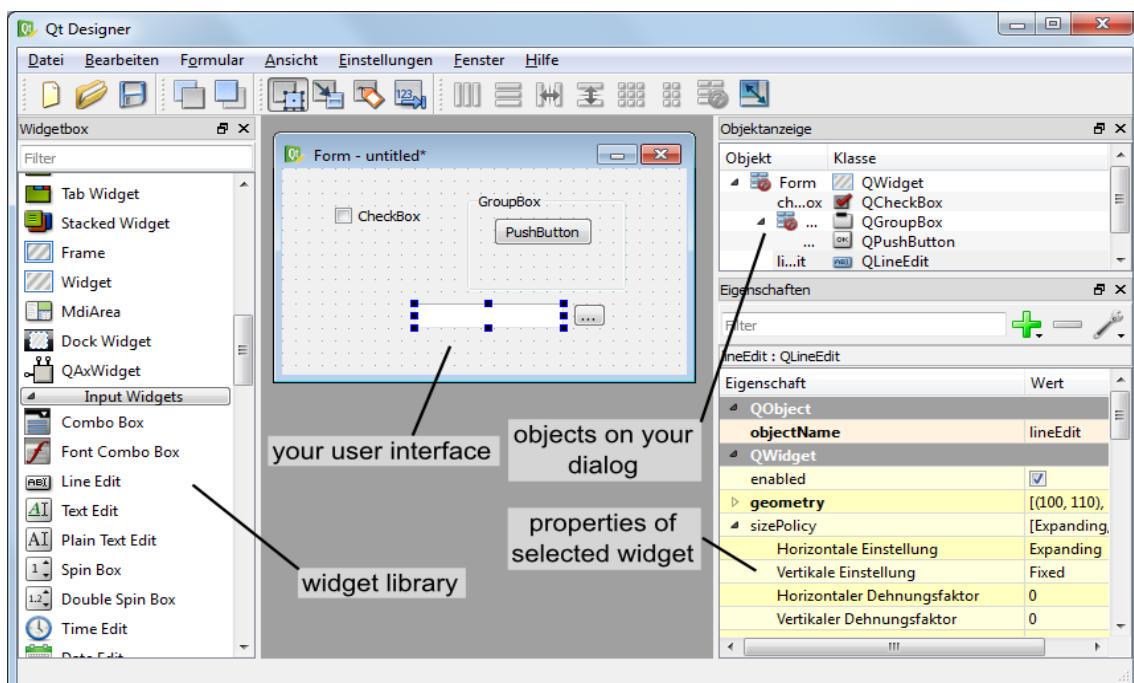


FIGURE 3.11 – Qt designer

Qt Designer est un outil graphique intégré à la suite de développement Qt, qui permet de concevoir des interfaces utilisateur graphiques de manière visuelle et intuitive. Il est utilisé en conjonction avec PyQt5 pour créer des interfaces pour des applications desktop.

Avec Qt Designer, vous pouvez glisser-déposer des widgets préconstruits tels que des boutons, des étiquettes, des zones de texte, des listes, des tableaux, etc., pour construire votre interface utilisateur. Il offre une interface conviviale avec une palette d'outils et une visionneuse en direct pour prévisualiser votre interface pendant que vous la concevez.

Ce même logiciel va ensuite générer un fichier «.ui» basé sur la norme XML. Ce fichier est utilisé par le compilateur Qt lors de la compilation du fichier «.pro». De plus, la génération des fichiers sources et des fichiers d'en-têtes est automatisée par Qt Designer à l'aide du compilateur intégré User Interface Compiler.

### 3.3.3 PyQt5 :



FIGURE 3.12 – PyQt5

PyQt5 est une bibliothèque Python qui permet de développer des interfaces graphiques pour des applications desktop. Elle est basée sur la bibliothèque Qt, qui est une bibliothèque de développement d'interfaces graphiques multiplateforme très populaire.

PyQt5 fournit un ensemble complet de fonctionnalités pour la création d'applications graphiques, y compris la gestion des fenêtres, des boîtes de dialogue, des boutons, des listes, des tableaux, des graphiques, etc. Elle offre également la possibilité de personnaliser l'apparence des interfaces en utilisant des feuilles de style, permettant ainsi de créer des interfaces visuellement attrayantes.

### 3.3.4 Pandas :

La bibliothèque logicielle open-source Pandas est spécifiquement conçue pour la manipulation et l'analyse de données en langage Python. Elle est à la fois performante, flexible et simple d'utilisation.

Grâce à Pandas, le langage Python permet enfin de charger, d'aligner, de manipuler ou encore de fusionner des données. Les performances sont particulièrement impressionnantes quand le code source back-end est écrit en C ou en Python.



FIGURE 3.13 – Pandas

### 3.3.5 VS Code :



FIGURE 3.14 – VS Code

Visual Studio Code est un éditeur de code extensible développé par Microsoft pour Windows, Linux et macOS2.

Les fonctionnalités incluent la prise en charge du débogage, la mise en évidence de la syntaxe, la complétion intelligente du code, les snippets, la refactorisation du code et Git intégré. Les utilisateurs peuvent modifier le thème, les raccourcis clavier, les préférences et installer des extensions qui ajoutent des fonctionnalités supplémentaires.

Le code source de Visual Studio Code provient du projet logiciel libre et open source VSCode de Microsoft publié sous la licence MIT permissive, mais les binaires compilés constituent un freeware, c'est-à-dire un logiciel gratuit pour toute utilisation mais privateur.

Dans le sondage auprès des développeurs réalisé par Stack Overflow en 2021, Visual Studio Code a été classé comme l'outil d'environnement de développement le plus populaire, avec 71,06 % des 82 277 répondants déclarant l'utiliser3.

### 3.3.6 Arduino IDE :

The Arduino integrated development environment (IDE) is a cross-platform application (for Microsoft Windows, macOS, and Linux) that is written in the Java programming language. It originated from the IDE for the languages Processing and Wiring. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and

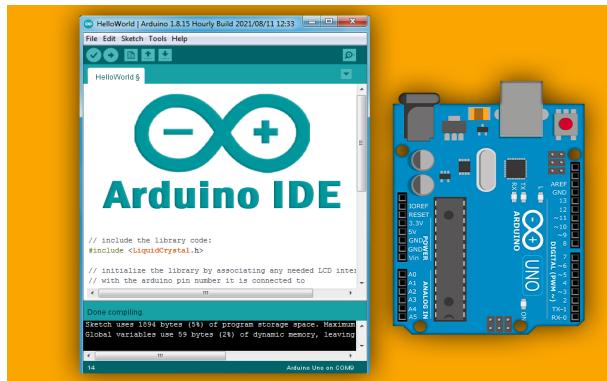


FIGURE 3.15 – Arduino IDE

syntax highlighting, and provides simple one-click mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub `main()` into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program `avrdude` to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

### 3.4 Développement des maquettes de l'application

L'application desktop que nous avons développée en utilisant Python et PyQt5 offre une interface conviviale et intuitive pour les utilisateurs, tels que Figure 3.17 montre l'interface principale où les médecins ou l'infirmière peuvent se connecter à leur compte à l'aide de nom de l'utilisateur et son mot de passe qui sera genre par l'administrateur de l'application comme montre la figure 3.18.



FIGURE 3.16 – Home Interface



FIGURE 3.17 – Interface Principale

La figure 3.19 est une interface qui contient tous les fonctions que peut faire l'utilisateur ; ajouter des nouveaux patients (Fig 3.20) , modifier ou supprimer leurs données (Fig 3.21 ), et le plus important c'est la consultation de ces patients (Fig 3.22) et les superviser (Fig 3.23)

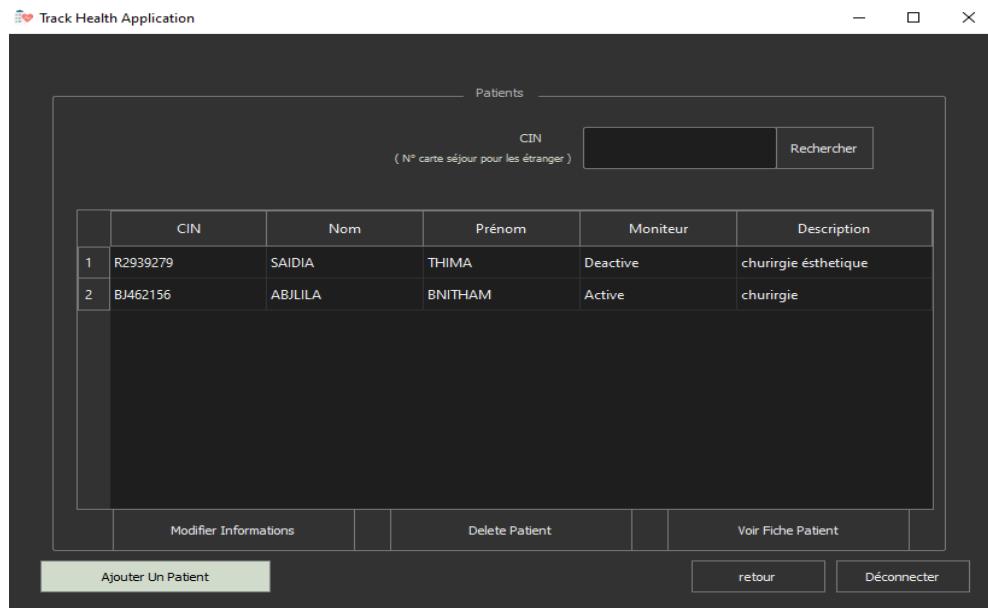


FIGURE 3.18 – Interface de connexion

[Np, 2020] Lorsque les utilisateurs sélectionnent un patient spécifique dans le tableau (Figure 3.19), ils sont dirigés vers une interface dédiée à ce patient (Figure 3.21). Cette interface affiche des informations plus détaillées sur le patient, telles que son historique médical, ses antécédents et ses résultats de tests précédents. De plus, si le moniteur portable est activé pour ce patient, un bouton de supervision à distance apparaît. En cliquant sur ce bouton, les utilisateurs peuvent accéder aux mesures en temps réel de la fréquence cardiaque, de la saturation en oxygène ( $SpO_2$ ) et de la température du patient. Ces mesures sont affichées en temps réel, permettant aux utilisateurs de surveiller l'état du patient et de réagir rapidement en cas de variations critiques.

Grâce à cette interface utilisateur bien conçue, les professionnels de la santé peuvent gérer facilement leurs patients, accéder rapidement aux informations médicales pertinentes et effectuer un suivi à distance précis et en temps réel. Cela facilite la prise de décisions éclairées et la fourniture de soins de qualité aux patients, même en dehors de l'environnement de réanimation traditionnel.

Track Health Application

Fiche Patient

ID	.....	N° Cin*	..... ( N° carte séjour pour les étranger )
Nom*	.....	Sexe*	Homme
Prénom*	.....	Nationalité*	Marocain
Description Court*			
Description Maladie			
 <input type="button" value="Import"/>			
<span>Moniteur</span> <input type="button" value="Active"/>			
<input type="button" value="Valider"/>			

\* Champs Obligatoire

FIGURE 3.19 – Interface des Patients

Track Health Application

Fiche Patient

ID	22242338	N° Cin*	R2939279 ( N° carte séjour pour les étranger )
Nom*	SAIDIA	Sexe*	Femme
Prénom*	THIMA	Nationalité*	Marocain
Description Court*		churrgie ésthetique	
Description Maladie		churrgie ésthetique pour visage ....	
 <input type="button" value="Modifier"/>			
<span>Moniteur</span> <input type="button" value="Deactive"/> <input type="button" value="Active"/> <input type="button" value="Deactive"/> <input style="background-color: orange; color: white; border-radius: 10px; padding: 2px 10px;" type="button" value="Moniteur"/>			
<input type="button" value="Valider"/>			

\* Champs Obligatoire

FIGURE 3.20 – Interface de modification

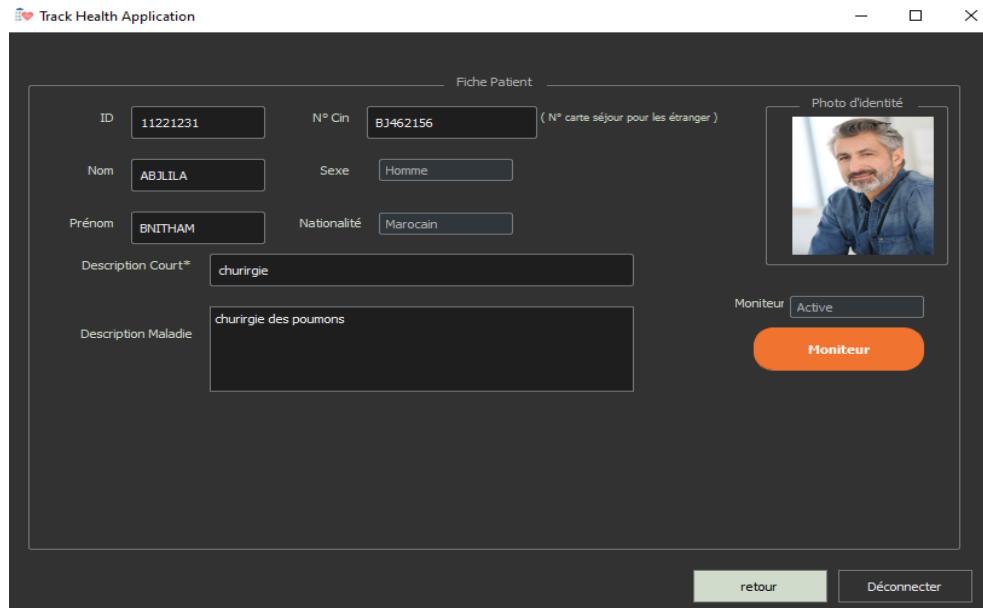


FIGURE 3.21 – Interface de consultation

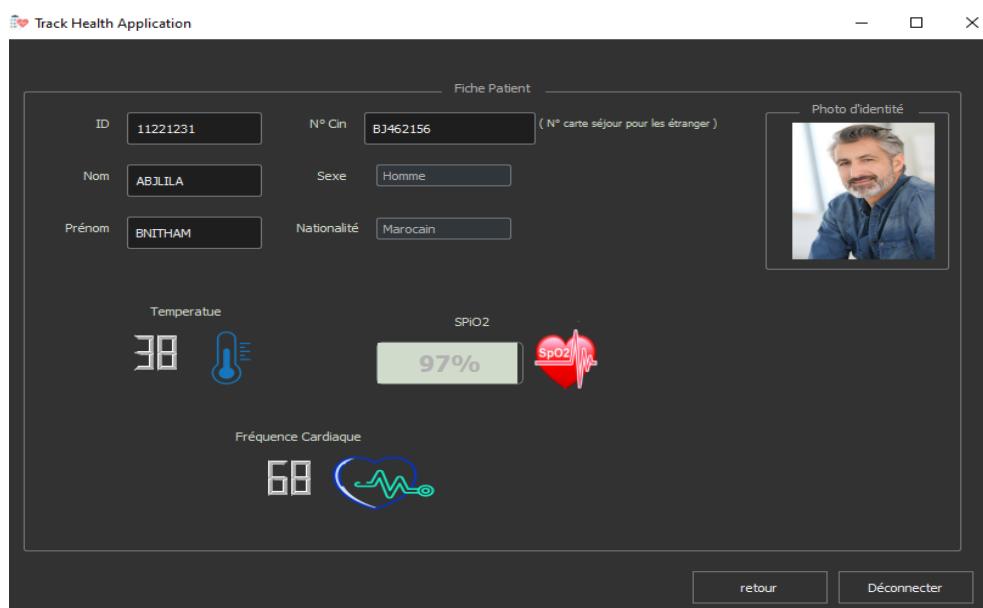


FIGURE 3.22 – Interface de supervision

## Conception de la base de données

Dans le cadre de la conception de la base de données pour notre application, nous avons utilisé des fichiers CSV locaux comme source de données et avons mis en place une API utilisant la bibliothèque Pandas pour faciliter les interactions entre l'application et la base de données.



FIGURE 3.23 – CSV

Nous avons créé deux tableaux principaux pour stocker les informations pertinentes. Le premier tableau (Figure 3.25) est dédié aux utilisateurs de l'application, tels que les médecins et les infirmières. Les colonnes de ce tableau incluent le nom, le prénom, le nom d'utilisateur et le mot de passe de chaque utilisateur. Cela permet de garantir un accès sécurisé à l'application et d'identifier chaque utilisateur de manière unique.

Le deuxième tableau (Figure 3.26) est spécifiquement conçu pour stocker les données des patients. Il comprend des colonnes telles que le nom, le prénom, le numéro de carte d'identité, l'identifiant à l'hôpital, le sexe, la nationalité, le lien vers l'image du patient, une description de la maladie et l'identifiant du médecin responsable. De plus, nous avons ajouté une colonne pour le statut du moniteur, qui indique si le patient est équipé d'un moniteur portable et actif pour la surveillance à distance. Cette structure nous permet d'organiser de manière efficace les informations sur les patients et de faciliter leur recherche et leur suivi.

Grâce à l'API basée sur Pandas, notre application est capable de lire, écrire et mettre à jour les données dans ces fichiers CSV locaux. Cela offre une flexibilité et une facilité de gestion, permettant aux utilisateurs d'ajouter de nouveaux patients, de modifier les informations existantes et de récupérer les données nécessaires pour la supervision à distance. L'utilisation des fichiers CSV en tant que base de données nous a offert une solution simple et efficace pour stocker les informations de manière structurée, sans nécessiter une infrastructure de base de données plus complexe.

En résumé, notre approche de conception de base de données en utilisant des fichiers CSV et une API basée sur Pandas nous a permis de gérer les informations des utilisateurs et des patients de manière organisée et efficace. Cela garantit la disponibilité des données nécessaires à l'application, tout en offrant une simplicité d'implémentation et de manipulation pour les différentes fonctionnalités de notre système de surveillance à distance des patients.

**CSV :** Comma-separated values, connu sous le sigle CSV, est un format texte ouvert représentant des données tabulaires sous forme de valeurs séparées par des virgules. Ce format n'a jamais vraiment fait l'objet d'une spécification formelle. Toutefois, la RFC 41801 décrit la forme la plus courante et établit son type MIME « text/csv », enregistré auprès de l'IANA.

	A	B	C	D
1	usernames	passwords	Nom	Prenom
2	Zak	123456789	doctor	smith
3	admin	Ziko0384	doctor	test
4	admin	123	Pr	zak

FIGURE 3.24 – Tableau base de données 1

A	B	C	D	E	F	G	H	I	J	K
id	cin	nom	prenom	moniteur_statut	sexe	nationalite	desc_court	desc_maladie	img_name	Id_medcin
11221231	BJ462156	ABJLILA	BNITHAM	Active	Homme	Marocain	chururgie	chururgie des p	11221231BJ462156	2
22242338	R2939279	SAIDIA	THIMA	Deactive	Femme	Marocain	chururgie esthetiqu	chururgie esthe	22242338R2939279	2

FIGURE 3.25 – Tableau base de données 2

### 3.5 Développement de Moniteur

Dans cette étape de développement, nous nous sommes concentrés sur la mise en place de la carte ESP8266 et d'Arduino pour mesurer les niveaux de SpO2, la fréquence cardiaque et la température à l'aide des capteurs MAX30100 et LM35. Ces mesures ont été affichées dans la console du moniteur série et ensuite transmises à un broker MQTT pour une transmission sécurisée et en temps réel.

Pour commencer, nous avons configuré la carte ESP8266 et Arduino en utilisant les bibliothèques appropriées pour établir la communication avec les capteurs MAX30100 (Figure 3.27) et LM35 (Figure 3.28). Le capteur MAX30100 a été utilisé pour mesurer les niveaux de SpO2 et la fréquence cardiaque, tandis que le capteur LM35 a été utilisé pour mesurer la température du patient.

Nous avons écrit le code nécessaire pour récupérer les valeurs des capteurs et les afficher dans le moniteur série. Cela nous a permis de vérifier la précision des mesures et de s'assurer que les capteurs fonctionnaient correctement.

Ensuite, nous avons intégré la fonctionnalité de transmission des données à un broker MQTT. Nous avons utilisé une bibliothèque MQTT pour établir une connexion sécurisée entre la carte ESP8266 et le broker MQTT. Les mesures de SpO2, de fréquence cardiaque et de température ont été encapsulées dans des messages MQTT et envoyées au broker pour une transmission efficace.

Cette approche nous a permis de collecter les données de manière fiable et de les transmettre en temps réel à la plateforme de surveillance à distance. Les données étaient ainsi disponibles pour être traitées et affichées sur l'application desktop, permettant aux professionnels de la santé de surveiller en continu l'état des patients.

En résumé, en utilisant la carte ESP8266 et Arduino avec les capteurs MAX30100 et LM35, nous avons réussi à mesurer les niveaux de SpO<sub>2</sub>, la fréquence cardiaque et la température des patients. Les valeurs ont été affichées dans le moniteur série et transmises à un broker MQTT pour une transmission sécurisée et en temps réel. Cette étape était cruciale pour la collecte précise des données et leur intégration dans l'application de surveillance à distance.

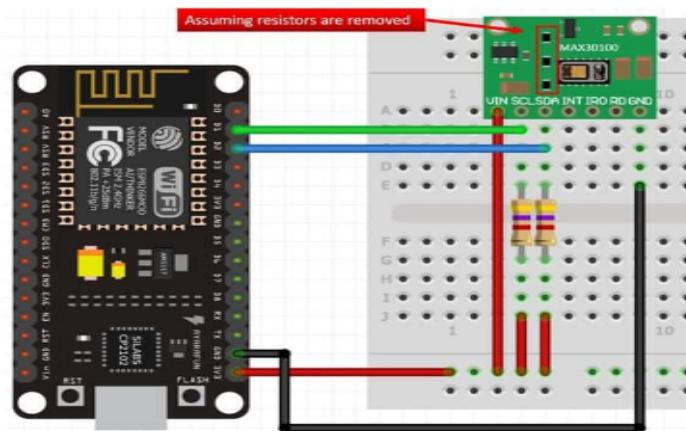


FIGURE 3.26 – connexions entre max30100 et l'ESP8266

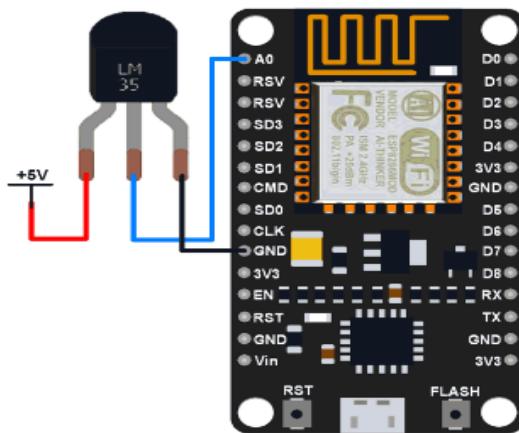


FIGURE 3.27 – connexions entre lm35 et l'ESP8266

[res, 2018] Dans notre développement, nous avons ajouté des fonctionnalités supplémentaires à la carte ESP8266 pour améliorer l'expérience utilisateur et renforcer les alarmes en cas de situations critiques. Nous avons intégré un buzzer au pin D5 de l'ESP8266 pour générer des codes sonores distincts en cas d'alarmes, et une LED bleue au pin D5 pour créer un clignotement proportionnel à la fréquence cardiaque du patient.

Le buzzer a été programmé pour émettre différents types de signaux sonores, tels que des bips continus ou des séquences de bips, afin d'attirer l'attention des utilisateurs lorsque des seuils critiques sont dépassés. Par exemple, si la saturation en oxygène (SpO<sub>2</sub>) descend en dessous d'un certain seuil ou si la fréquence cardiaque dépasse une valeur prédéfinie, le buzzer se déclenche pour alerter immédiatement les professionnels de la santé.

En parallèle, nous avons connecté une LED bleue au pin D8 de l'ESP8266. Cette LED a été programmée pour clignoter à une fréquence proportionnelle à la fréquence cardiaque du patient. Ainsi, plus la fréquence cardiaque est élevée, plus la LED bleue clignote rapidement. Cela offre une indication visuelle claire de l'état cardiaque du patient et permet une évaluation rapide de l'urgence de la situation.

Ces fonctionnalités supplémentaires, le buzzer et la LED bleue, renforcent la capacité de notre système à attirer l'attention des professionnels de la santé en cas de situations critiques. Ils fournissent des signaux sonores et visuels distincts, permettant une réactivité immédiate face à des conditions potentiellement dangereuses. Cette amélioration globale de l'interface utilisateur et des alarmes contribue à renforcer la sécurité et la qualité des soins fournis aux patients surveillés à distance.

## 3.6 Connectivité MQTT

```
// WiFi
const char *ssid = ""; // Enter your WiFi name
const char *password = ""; // Enter WiFi password

// MQTT Broker
const char *mqtt_broker = "test.mosquitto.org";
const char *topic = "monitorAlert/CHU";
const char *mqtt_username = "";
const char *mqtt_password = "";
const int mqtt_port = 1883;

WiFiClient espClient;
PubSubClient client(espClient);
```

FIGURE 3.28 – code de connexion C/C++ mqtt

Nous avons utilisé la bibliothèque Paho MQTT pour assurer la connectivité MQTT entre notre application desktop et le broker Mosquitto. Cette étape était essentielle pour permettre la transmission des données entre la carte ESP8266 et l'application, en utilisant le protocole MQTT sécurisé.

Nous avons configuré notre application desktop pour se connecter au broker Mosquitto en utilisant les informations d'adresse IP et de port appropriées. Une fois la connexion établie, notre application s'est abonnée à trois topics spécifiques pour recevoir les valeurs mesurées par les capteurs de la carte

ESP8266.

Ces trois topics étaient configurés pour recevoir les mesures de la fréquence cardiaque, de la saturation en oxygène (SpO2) et de la température. À mesure que la carte ESP8266 publiait les données sur ces topics, notre application les recevait instantanément.

Une fois les données reçues, nous les avons traitées et affichées sur l'interface utilisateur de l'application. Cela permettait aux médecins et aux infirmières d'avoir une vue en temps réel des mesures des patients, les aidant ainsi à prendre des décisions éclairées concernant leur surveillance et leurs soins.

La connectivité MQTT nous a permis d'établir une communication fiable et sécurisée entre la carte ESP8266 et l'application desktop. Grâce à cette architecture, nous avons pu garantir la transmission rapide et efficace des données, assurant ainsi une surveillance en temps réel des patients à distance.

En résumé, en utilisant la bibliothèque Paho MQTT, nous avons établi une connexion entre notre application desktop et le broker Mosquitto. Notre application s'est abonnée à trois topics pour recevoir les données de fréquence cardiaque, de SpO2 et de température publiée par la carte ESP8266. Cette connectivité MQTT a joué un rôle clé dans l'établissement d'une communication fiable et sécurisée, permettant aux professionnels de la santé de surveiller les patients à distance en temps réel.

```
class MqttApp(QThread):
    spio2 = pyqtSignal(str)
    temp = pyqtSignal(str)
    pulse_rate = pyqtSignal(str)

    def run(self):
        self.client = paho.Client()
        self.client.on_connect = self.on_connect
        self.client.on_message = self.on_message
        self.client.on_publish = self.on_publish
        self.client.connect("test.mosquitto.org", 1883, 60)
        self.client.loop_forever()

    def on_connect(self, client, userdata, flags, rc):
        self.client.subscribe("moniteurCHU/temp")
        self.client.subscribe("moniteurCHU/spio2")
        self.client.subscribe("mntrCHU/plsRate")

    def on_message(self, client, userdata, msg):
        topic, message = msg.topic, msg.payload.decode("utf-8")
        print(topic + " -> " + str(message))
        if topic == "moniteurCHU/temp":
            self.temp.emit(str(message))
        elif topic == "moniteurCHU/spio2":
            self.spio2.emit(message)
        elif topic == "mntrCHU/plsRate":
            self.pulse_rate.emit(message)
```

FIGURE 3.29 – code de connexion python mqtt

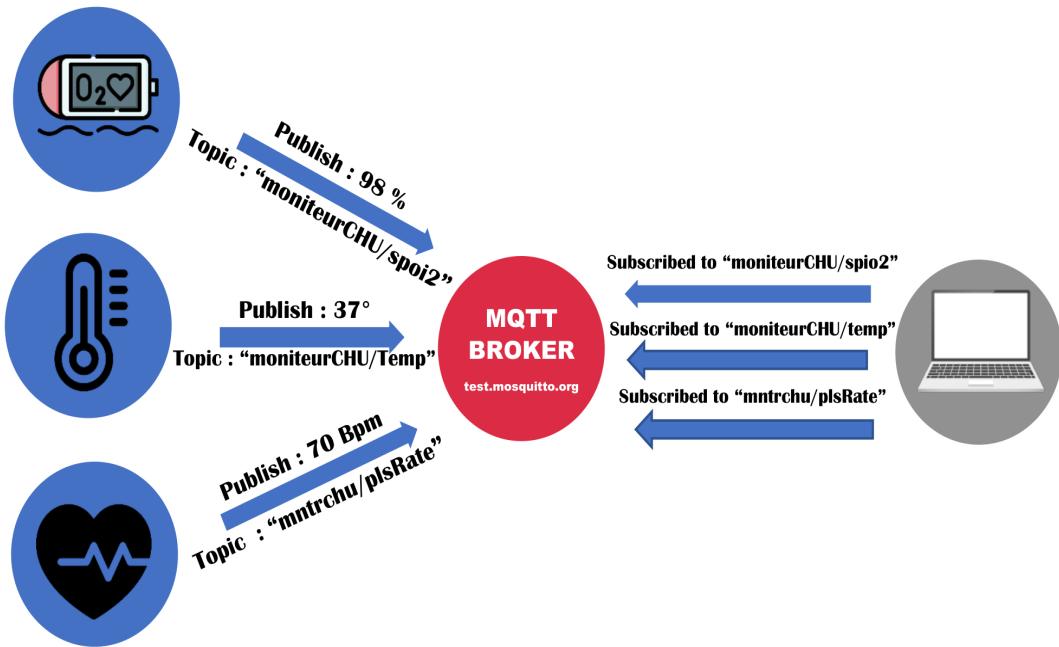


FIGURE 3.30 – Architecture MQTT

### 3.7 Concevoir la disposition du PCB

Dans la phase de conception du PCB (Printed Circuit Board), nous avons utilisé le logiciel EasyEDA pour concevoir le schéma électrique et le routage des pistes. EasyEDA nous a permis de créer un design personnalisé pour notre PCB, en plaçant les composants de manière optimale et en traçant les connexions nécessaires.

Cependant, nous avons rencontré des contraintes en termes de fabrication du PCB. Nous n'avions pas les moyens ou l'accès à une imprimante spécifique pour imprimer notre PCB. Nous avons donc opté pour une alternative en utilisant une platine de prototypage universelle double face de dimensions 5x7 cm.

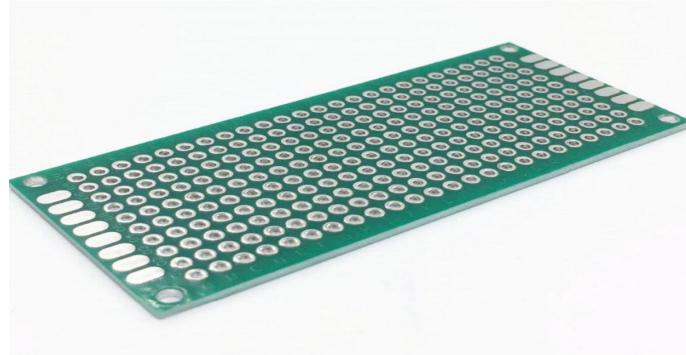


FIGURE 3.31 – platine de prototypage

Nous avons soudé les composants électroniques directement sur cette platine de prototypage. Cette approche nous a permis de tester et de valider notre circuit électronique sans avoir à fabriquer un PCB spécifique. Bien que cela ne soit pas une solution permanente, cela a été suffisant pour la phase de développement et de validation de notre prototype.

En utilisant cette platine de prototypage, nous avons soigneusement soudé les composants tels que le microcontrôleur, les capteurs et les éléments de connectivité. Nous avons veillé à suivre les schémas de câblage et à établir les connexions appropriées entre les différents composants.

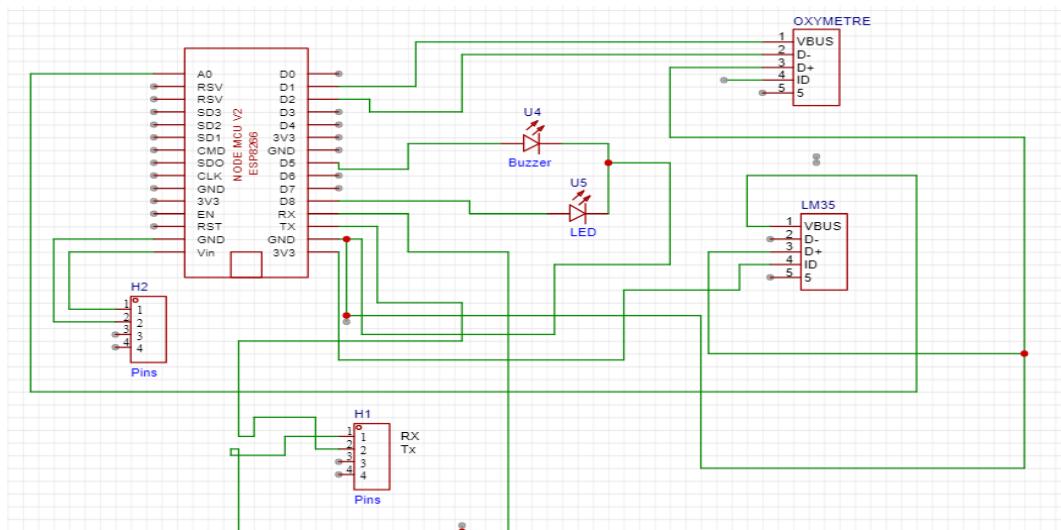


FIGURE 3.32 – Schéma de circuit

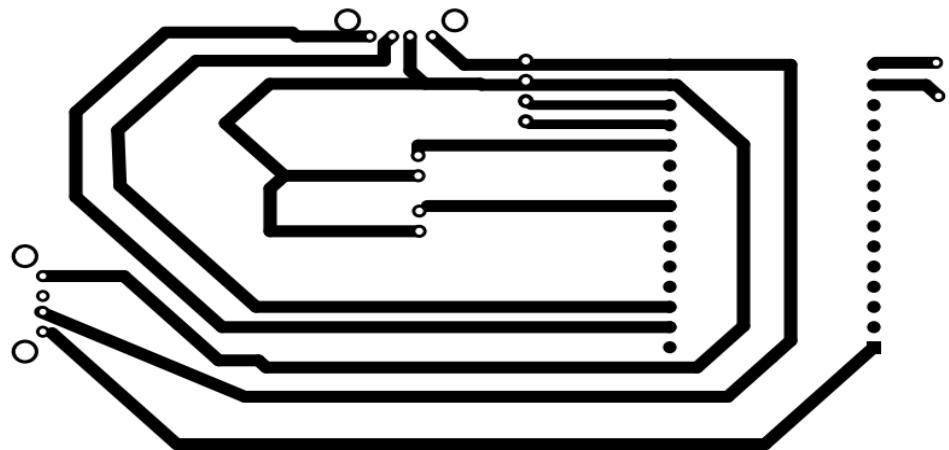


FIGURE 3.33 – PCB à imprimer

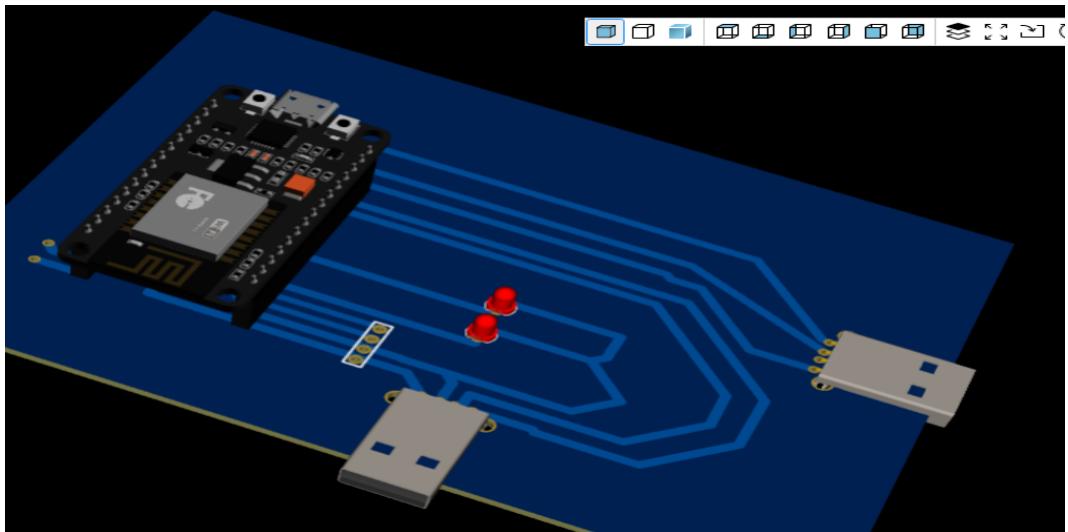


FIGURE 3.34 – PCB 3D

En résumé, en raison de contraintes de fabrication du PCB, nous avons utilisé une platine de prototypage universelle double face de dimensions 5x7 cm pour souder les composants de notre circuit électronique. Cette approche nous a permis de tester et de valider notre prototype tout en travaillant avec des ressources limitées.

## 3.8 Conception De Boîtier

Pour la conception du boîtier du moniteur, nous avons opté pour l'utilisation d'une boîte étanche d'électricité de dimensions 15 cm x 10 cm. Cette taille généreuse nous permet d'accueillir notre PCB ainsi que l'Arduino Uno, qui agit comme la carte graphique du moniteur.

À l'intérieur du boîtier, nous avons positionné le PCB de manière à maximiser l'espace disponible et à faciliter le câblage. Nous avons soigneusement disposé les composants tels que l'écran LCD I2C, le buzzer et les deux LED.

La première LED, de couleur orange, est utilisée pour indiquer que le moniteur est allumé et opérationnel. Elle fournit une indication visuelle claire pour s'assurer que le moniteur est prêt à être utilisé.

La deuxième LED est utilisée pour suivre la fréquence cardiaque du patient. Cette LED clignote en fonction du rythme cardiaque détecté, fournissant ainsi une représentation visuelle de la fréquence cardiaque en temps réel.

La conception du boîtier vise à offrir une solution pratique et durable. La boîte étanche protège le circuit électronique contre les dommages causés par les éléments extérieurs tels que l'humidité et la poussière, assurant ainsi un fonctionnement fiable dans divers environnements.

la conception du boîtier du moniteur repose sur l'utilisation d'une boîte étanche d'électricité de taille adaptée, permettant d'abriter efficacement le PCB, l'Arduino Uno, l'écran LCD, le buzzer et les LED. Cette conception garantit une protection adéquate du circuit électronique tout en offrant une interface visuelle claire pour le suivi de la fréquence cardiaque.

En plus des fonctionnalités mentionnées précédemment, nous avons également intégré des indications sonores dans le boîtier du moniteur. Ces indications sonores sont générées à l'aide d'un buzzer et servent à transmettre des informations supplémentaires aux utilisateurs.

Plus précisément, nous avons programmé le buzzer pour émettre deux bips sonores en cas d'erreur. Cette indication sonore est déclenchée lorsqu'une erreur ou une situation anormale est détectée, telle qu'une défaillance de capteur ou une perte de connexion.

En revanche, lorsque tout fonctionne correctement, le buzzer émet trois bips sonores pour signaler le succès. Cette indication sonore est utilisée pour confirmer que les mesures sont en cours, que les données sont transmises avec succès et que le moniteur fonctionne de manière optimale.

Ces indications sonores ajoutent une dimension supplémentaire à l'expérience utilisateur en fournissant des signaux audibles qui complètent les indicateurs visuels. Cela permet aux utilisateurs, notamment aux médecins et aux infirmières, de détecter rapidement les erreurs ou les réussites, même sans avoir à observer en permanence l'interface visuelle.

En résumé, le boîtier du moniteur comprend des indications sonores générées par un buzzer. Deux bips sonores signalent une erreur, tandis que trois bips sonores indiquent un succès. Ces indications sonores améliorent l'expérience en fournissant des signaux audibles pour compléter les indicateurs visuels, facilitant ainsi la détection rapide des problèmes et la confirmation des opérations réussies.

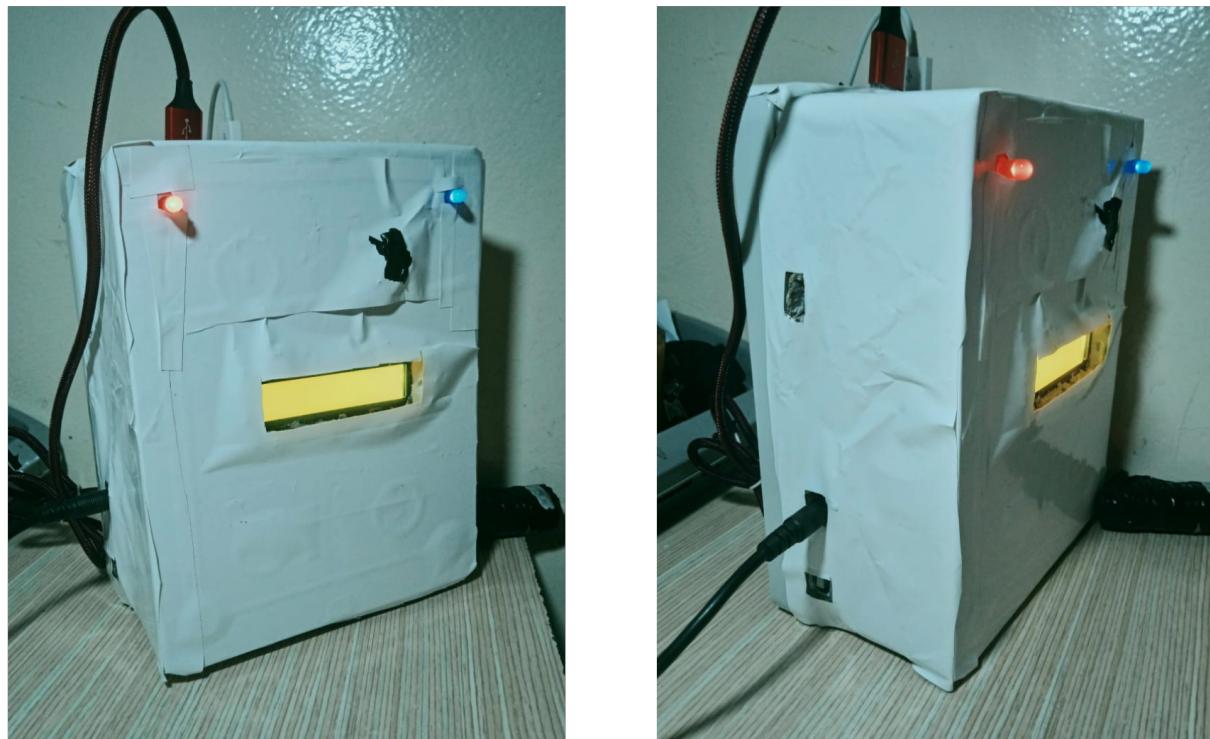


FIGURE 3.35 – résultat final du boîtier de moniteur



FIGURE 3.36 – intérieur du boîtier de moniteur

## Conclusion

Nous avons réussi à concrétiser la mise en œuvre de notre système de surveillance à distance des patients. Nous avons abordé la conception et le développement de l'application desktop utilisant Python et PyQt5, ainsi que la programmation de la carte ESP8266 pour la collecte des données des capteurs.

Grâce à notre travail, nous avons pu créer une interface conviviale pour les utilisateurs, offrant un accès sécurisé et intuitif aux données des patients. L'intégration des capteurs de saturation en oxygène, de fréquence cardiaque et de température nous a permis de collecter des informations vitales en temps réel, fournissant ainsi un suivi précis de l'état des patients.

Nous avons également mis en place la connectivité MQTT entre la carte ESP8266 et l'application desktop, permettant la transmission fluide des données vers le broker MQTT. Cela a assuré une communication fiable et rapide entre les différents composants du système.

De plus, la conception et la réalisation du PCB nous ont permis de consolider notre système dans un format compact et fonctionnel. La soudure des composants sur la platine de prototypage a été une étape importante pour garantir le bon fonctionnement de notre dispositif.

Nous avons présenté la conception du boîtier du moniteur, qui offre une solution pratique et fonctionnelle pour abriter les composants électroniques essentiels. Cette conception témoigne de l'efficacité et de la polyvalence de notre moniteur de surveillance à distance. En combinant des mesures précises, une interface conviviale et des fonctionnalités supplémentaires telles que les indications sonores, notre moniteur offre une solution complète pour la surveillance à distance des patients.

Ce chapitre a détaillé les étapes de réalisation de notre projet, mettant en évidence à la fois les aspects matériels et logiciels. Nous sommes parvenus à développer avec succès une application desktop fonctionnelle et un petit moniteur portable pour la surveillance à distance des patients. Ces réalisations constituent une base solide pour la suite de notre projet et ouvrent la voie à de nombreuses opportunités d'amélioration et de développement futur.

Tous les sources code développées dans ce projet côté moniteur et application desktop se trouvent à Github -> [www.github.com/zakariarhiba/PFE\\_TIMB](https://www.github.com/zakariarhiba/PFE_TIMB) <- [Cod, ].

# Conclusion générale

En conclusion, ce rapport de fin d'étude a présenté le développement d'une application IoT pour la surveillance à distance des patients. Nous avons identifié le besoin de suivi continu des patients dans les unités de réanimation où les lits médicaux et les moniteurs sont limités. Notre projet propose une solution innovante en utilisant une petite moniteur portable connecté à une application desktop.

L'un des points forts de notre projet réside dans la capacité à surveiller en temps réel les signes vitaux des patients tels que la fréquence cardiaque, la saturation en oxygène et la température. Cette surveillance à distance permet aux médecins et aux infirmières de suivre de manière continue l'état des patients, même lorsqu'ils sont éloignés physiquement.

De plus, notre application offre la possibilité d'ajouter des fonctionnalités supplémentaires à l'avenir. Par exemple, l'intégration de l'intelligence artificielle pourrait permettre de générer des indicateurs et des statistiques sur l'état des patients à partir des données stockées dans la base de données. Cela fournirait une analyse approfondie et des informations précieuses pour les professionnels de la santé, facilitant ainsi la prise de décisions éclairées.

Nous envisageons également d'ajouter des actionneurs tels que des messages d'alerte envoyés aux médecins en cas de seuils critiques atteints. Cela permettrait une intervention rapide et efficace, garantissant des soins adaptés et une surveillance proactive des patients.

En résumé, notre projet a démontré la faisabilité et l'efficacité de la surveillance à distance des patients en utilisant une application desktop connectée à un petit moniteur portable. Ce rapport de fin d'étude souligne les avantages de cette approche, tels que la continuité des soins, la flexibilité et la possibilité d'ajouter des fonctionnalités supplémentaires à l'avenir. Ce projet ouvre la voie à de nouvelles avancées dans le domaine de la surveillance médicale et contribue à l'amélioration des soins aux patients.

Ce projet a été vivant, entraînant et motivant pour la suite de nos études. Nous pensons avoir entraperçu une partie de notre future vie active.

# Bibliographie

- [Cod, ] Le dossier git ([www.github.com/zakariarhiba/pfe<sub>t</sub>imb](http://www.github.com/zakariarhiba/pfe_timb)) contient tout le code développé dans ce projet.
- [mic, ] Max30100 pulse oximeter and heart rate sensor with esp8266.
- [Pyq, 2016] (2016). *PyQt5 Tutorial Documentation*. Andrew Steele.
- [res, 2018] (2018). *IOT based controlling of hybrid energy system using ESP8266*. Institute of Electrical and Electronics Engineers.
- [Ap, 2018] (2018). *Service composition approaches in IoT : A systematic review*. Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.
- [Np, 2020] (2020). *A Paho MQTT based IoT-RFID Attendance System using NodeMCU Firmware*. International Research Journal of Engineering and Technology, M. Tech. Student (Dept. of ETC, DIEMS, Aurangabad, Maharashtra, India).
- [Çetin, ] Çetin, M. A. . S. H. E. *Healthcare and patient monitoring using IoT*. Bolu Abant Izzet Baysal University.