

République Algérienne Démocratique et Populaire  
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique  
Université Mohamed El Bachir El Ibrahim de Bordj Bou Arreridj  
Faculté de Mathématiques et Informatique  
Département d'informatique



## **Mémoire**

Présenté pour l'obtention du diplôme.

## **Master en Informatique**

Spécialité : Réseaux et multimédia

## **THÈME**

Réalisation d'un Framework pour la fusion et l'échange d'informations dans les réseaux de capteurs.

Présenté par :

- NOUI SEIF EDDINE
- RAHALI AHMED SAMI

Membres de jury :

- Présidente : Manal LAMRI
- Examineur : Mourad NOUIOUA

Encadrant :

- SABRI LYAZID

2023 /2024

# Dédicace

---

À mes très chers parents

J'ai toujours attendu avec une grande impatience ce jour où je vous témoignerai toute la gratitude d'un fils qui s'est toujours vanté de vous avoir comme père et mère.

Aucune dédicace ne peut vous exprimer la profondeur de mon amour, de mon appréciation de ma gratitude infinie pour tous les sacrifices que vous avez sincèrement consentis pour mon éducation et mes encouragements toutes ces années d'études. Vous avez guetté mes pas et vous m'avez couvert de tendresse, et vos prières et vos bénédictions m'ont été d'une grande aide pour mener à bien mes études.

Je vous serai fidèle toute ma vie et aucune déclaration ne me dégagera de la lourde responsabilité que je ressens à votre égard.

Je demande à Dieu le tout puissant de vous accorder santé et prospérité et vous accorde une longue vie afin que je puisse remplir votre rôle

Qu'Allah, le Très-Haut, fasse que le meilleur reste à venir.

# Remerciement

---

C'est pour nous un plaisir autant qu'un devoir de remercier toutes les personnes qui ont pu contribuer de près ou de loin à l'établissement de ce projet, qui nous ont aidé, soutenu et ont fait en sorte que ce travail ait eu lieu.

Ainsi, nous exprimons notre gratitude et nous tenons à remercier M. Sabri Lyazid qui nous a encadré et qui n'a épargné aucun effort pour nous orienter afin qu'on puisse mener à bien ce projet.

Nous tenons également à remercier les membres du jury pour le temps précieux qu'ils nous ont accordé à l'étude de notre mémoire.

Enfin, nous remercions affectueusement nos parents, qui ont toujours su nous faire confiance et nous soutenir sans compter dans nos études.

# Résumé

---

L'Internet des objets (IdO) est un réseau de dispositifs physiques intégrés avec des capteurs, des logiciels et d'autres technologies pour échanger des données avec d'autres dispositifs et systèmes sur Internet et des réseaux informatiques. Afin d'assurer cette communication, l'IdO fait appel à différents protocoles de communication qui vont permettre à ses objets de communiquer. Parmi ses protocoles, celui qui est considéré par le consortium mondial, Organization for the Advancement of Structured Information Standards (OASIS) comme un standard, le protocole MQTT. Toutefois, ces protocoles sont constamment développés pour permettre une meilleure interaction, et chaque protocole possède ses propres fonctionnalités et performances spécifiques.

Le travail présenté dans ce rapport s'inscrit dans ce cadre et consiste à créer une application web/mobile pour contrôler l'infrastructure IdO telle qu'une maison intelligente (en anglais, smart home). La première partie est constituée de l'état de l'art de la technologie IdO, des protocoles applicatifs les plus répandus et plus utilisés en domotique et quelques exemples de solutions actuelles. La deuxième partie est réservée à l'analyse des besoins et la conception détaillée de notre solution. La dernière partie décrit le choix des outils techniques et la mise en œuvre de notre solution, ainsi que son déploiement dans un environnement de production.

- o **Mots Clés :** Internet des Objets, protocoles de communication, MQTT, maison intelligente.

# Abstract

---

The Internet of Things (IoT) is not just a network of physical devices integrated with sensors, software, and other technologies. A revolutionary concept enables these devices to exchange data with other devices and systems over the Internet and computer networks. The backbone of this communication is the IoT's diverse range of communication protocols, each with unique features and performance. Among them is the MQTT protocol, a standard recognized by the Organization for the Advancement of Structured Information Standards (OASIS). These protocols are not static; they are in a constant state of evolution, always striving to facilitate better interaction and underscoring the dynamic nature of this field.

The work presented in this report is an innovative endeavor that involves creating a web/mobile application to control IoT infrastructure, such as a smart home. The first part delves into state-of-the-art IoT technology, the most widespread and most used application protocols in home automation, and some examples of current solutions. The second part is a meticulous analysis and detailed design of our solution. The last part is the culmination of our efforts, describing the choice of technical tools and the implementation of our solution, as well as its deployment in a production environment.

- o **Keys:** Internet of Things, communication protocols, MQTT, Smart home.

## ملخص

إنترنت الأشياء عبارة عن شبكة من الأجهزة المادية المدمجة مع أجهزة الاستشعار والبرامج والتقنيات الأخرى لتبادل البيانات مع الأجهزة والأنظمة الأخرى عبر الإنترنت وشبكات الكمبيوتر. ومن أجل ضمان هذا الاتصال، يستخدم إنترنت الأشياء بروتوكولات اتصال مختلفة، والتي ستسمح لكائناته بالتواصل؛ من بين بروتوكولاته هناك عدة بروتوكولات، بين البروتوكولات لدينا بروتوكول "إم كيو تي تي". يتم تطوير هذه البروتوكولات باستمرار لتمكين التفاعل بشكل أفضل، ولكل بروتوكول ميزاته وأدائه الخاص.

يقع العمل المقدم في هذا التقرير ضمن هذا الإطار ويتكون من إنشاء تطبيق ويب/هاتف محمول للتحكم في البنية التحتية لإنترنت الأشياء (المنزل الذكي). يتكون الجزء الأول من أحدث تقنيات إنترنت الأشياء وبروتوكولات التطبيقات الأكثر انتشاراً. يشجع استخدام الأتمتة في المنزل وبعض الأمثلة على الحلول الحالية. الجزء الثاني مخصص لتحليل الاحتياجات والتصميم التفصيلي لحلنا. يصف الجزء الأخير اختيار الأدوات التقنية وتنفيذ الحل الذي نقدمه، بالإضافة إلى نشره في بيئة الإنتاج.

○ المفاتيح: إنترنت الأشياء، بروتوكولات الاتصال "إم كيو تي تي"، المنزل الذكي.

## Table des matières

List des figures.....	10
Liste des tableaux.....	11
Liste des abréviations.....	12
Chapitre 01 Introduction Générale.....	1
1.1. Contexte.....	1
1.2. Objectifs.....	1
1.3. Méthodologie et résultats.....	1
1.4. Aperçu du rapport.....	2
Chapitre 02 Paradigme de IdO dans la vie du quotidien.....	3
2.1. Introduction.....	3
2.2. Architecteur.....	4
2.3. Plateforme Arduino.....	5
2.3.1. Arduino Fonctionnement.....	5
2.3.2.Energia.....	6
2.4. Maison intelligente dans l’histoire et le futur.....	7
2.5. Types de systèmes de maison intelligente.....	7
2.6. Maisons intelligentes en Algérie (Statistiques).....	8
2.7. Conclusion.....	8
Chapitre 03 Architecture propose MQTT.....	9
3.1. Introduction.....	9
3.2. Couche des dispositifs IdO.....	9
3.2.1. Les dispositifs d’une maison intelligente.....	9
3.2.1.1. Composants Principaux de la Couche de Dispositif IdO Capteurs.....	9
a) Les capteurs :.....	9
b) Actionneurs :.....	10
c) Microcontrôleurs et microprocesseurs :.....	10
d) Modules de Communication :.....	10
3.2.2. Considérations pour la Conception de la Couche de Dispositif IdO.....	10
3.2.3. Le scénario.....	11
Intégration et Fonctionnement.....	16
3.3. Couche de communication.....	18
3.3.1. LoRaWAN (long-range wide area network).....	18
3.3.1.1. Caractéristique de LoRaWAN.....	18
3.3.1.2. Architecture de LoraWAN le protocole réseau.....	18

3.3.2. Sigfox.....	19
3.3.2.1. Caractéristique de la technologie Sigfox.....	19
3.3.2.2. Les fréquences utiliser.....	20
3.3.2.3. Architecture et le fonctionnement de Sigfox.....	20
3.3.3. L'idée de full Duplex et les inconvénients de HTTP.....	21
3.3.3.1. Le Full Duplex.....	21
3.3.3.2. Inconvénient de http.....	21
3.3.4. Le modelé des services des messageries.....	22
3.3.4.1. Composant : du modèle des Services de messageries dans l'IdO.....	22
3.3.4.2. Les protocoles de Messagerie Utilisée dans l'IdO.....	22
3.3.4.3. Modèle de Publication/Abonnement (PUB/SUB).....	23
3.3.4.4. Sécurité dans les Services de Messageries <i>IdO</i> .....	24
3.3.5. MQTT.....	24
3.3.5.1. Transport de télémétrie de file d'attente de messages (MQTT).....	24
3.3.5.2. Le principe de MQTT.....	24
3.3.5.3. Fonctionnement de MQTT.....	25
3.3.5.4. Exemples d'Utilisation de MQTT.....	25
3.3.5.5. Rubrique MQTT.....	25
3.3.5.6. Publication MQTT.....	25
3.3.5.7. Abonnement MQTT.....	25
3.3.5.8. Sécurité dans MQTT.....	26
3.3.6. Le protocole MQTT.....	26
3.3.6.1. Concepts de base de MQTT.....	27
3.3.6.2. Sujets et abonnements.....	27
3.3.6.3. Messages conservés.....	28
3.3.6.4. Testaments.....	28
3.3.6.5. Keep Alive et prise de contrôle des clients.....	29
3.3.6.6. Messages de session persistante et de file d'attente.....	29
3.3.7. Mosquitto Broker (Serveur).....	29
CHAPITRE 04 Implémentation.....	31
4.1. INTRODUCTION.....	31
4.2 Les langages utilisés.....	32
4.2.1. Langage HTML (HyperText Markup Langage).....	32
Introduction.....	32
Éléments HTML de Base.....	32
Attributs HTML :.....	32
4.2.2. Langage (SCC).....	33



4.2.3. Langage (JS).....	33
Avantages de JavaScript.....	34
4.2.4. Langage (python).....	34
Avantages de Python.....	34
4.2.5. Bootstrap.....	35
4.2.6. MQTT Bibliothèque.....	36
4.3. Les diagrammes UML.....	36
4.3.1. La méthode utilisée dans la conception.....	36
Modélisation du système.....	36
4.3.2 Représentants du système et tâches.....	37
4.3.3. Diagramme de cas d'utilisation.....	38
4.3.4. Diagramme de classes.....	39
4.3.5. Diagramme de séquence.....	40
4.3.6. Diagramme d'activité.....	40
DéTECTEURS.....	42
4.4. Scénarios.....	46
4.5. Conclusion.....	47
Chapitre 05 Conclusion générale.....	48
Références :.....	49

## List des figures

Figure 1 Architecture IdO [20]	4
Figure 2 code arduino [22].	6
Figure 3 Vue générale de note application pour contrôler une maison.	11
Figure 4 capture sur le pc (Dashboard) le salon et la chambre d'enfant.	11
Figure 5 capture sur le smart phone le salon et la chambre d'enfant.	12
Figure 6 capture la cuisine dans le smart phone.	13
Figure 7 capture sur le pc (Dashboard) la salle de bain	13
Figure 8 capture sur le pc (Dashboard) de la cuisine.	14
Figure 9 capture sur le pc (Dashboard) du Hall et dans le smart phone.	15
Figure 10 capture sur le pc (Dashboard) du chambre d'enfant et dans le smart phone.	15
Figure 11 capture sur le pc (Dashboard) du garage et dans le smart phone.	16
Figure 12 capture sur le pc (Dashboard) de Smart home.	17
Figure 13 LoraWAN [5]	18
Figure 14 LoraWAN [6]	19
Figure 15 Sigfox [7]	20
Figure 16 L'idée de full Duplex et les inconvénients de HTTP [9]	22
Figure 17 Les protocoles de Messagerie Utilisée dans l'IdO [8]	23
Figure 18 Les protocoles de Messagerie MQTT et CoAP l'IOT [10]	24
Figure 19 Transport de télémétrie de file d'attente de messages [11]	26
Figure 20 IOT M2M [12]	26
Figure 21 Concept de base de MQTT [13]	27
Figure 22 Système de sécurité domestique. [40]	31
Figure 23 Fragment de code HTML.	32
Figure 24 Fragment de code CSS.	33
Figure 25 Fragment de code java script.	33
Figure 26 Fragment de code Python.	34
Figure 27 Fragment de code de bootstrap.	35
Figure 28 Diagramme de cas d'utilisation de la maison intelligente.	38
Figure 29 Classe de diagrammes	39
Figure 30 Diagramme de cas d'utilisation utilisateur.	40
Figure 31 Diagramme d'activité pour l'utilisateur.	41
Figure 32 Diagramme de séquence utilisateur.	41
Figure 33 Représente le schéma de cas d'utilisation des détecteurs.	42
Figure 34 Représente le diagramme d'activité des détecteurs.	43
Figure 35 Représente le diagramme de séquence des détecteurs.	43
Figure 36 Représente le diagramme d'activité de l'actionneur.	44
Figure 37 Diagramme d'activité de l'actionneur.	44
Figure 38 Diagramme de séquence d'actionneur.	45
Figure 39 Dispositifs réels mis en œuvre dans montre expérience	47

## Liste des tableaux

Tableau 1 Les avantages des LoraWAN et Sigfox.	21
Tableau 2 Les avantages de full Duplex et Les inconvénients de http	21
Tableau 3 Différent type de qualité de services [4]	28
Tableau 4 Représentants du système et tâches.	37

# Liste des abréviations

- **AMQP** : Advanced Message Queuing Protocol (Protocole de Mise en File d'Attente Avancée des Messages)
- **API** : Application Programming Interface (Interface de programmation d'applications)
- **Arduino** : Une plateforme de prototypage open-source basée sur du matériel et un logiciel flexible, utilisée pour le développement de projets électroniques.
- **CCS** : Code Composer Studio
- **CoAP** : Constrained Application Protocol (Protocole de Constrained Application)
- **CSS** : Cascading Style Sheets (Feuilles de style en cascade)
- **DDS** : Data Distribution Service (Service de Distribution de Données)
- **HTML** : HyperText Markup Language (Langage de Balisage Hypertexte)
- **HTTP/HTTPS** : HyperText Transfer Protocol / Secure (Protocole de Transfert Hypertexte / Protocole de Transfert Hypertexte Sécurisé)
- **IA** : Intelligence Artificielle (Artificial Intelligence)
- **IDE** : Integrated Development Environment (Environnement de Développement Intégré)
- **IdO** : Internet des Objets
- **IP** : Internet Protocol (Protocole Internet)
- **ISM** : Industriel, Scientifique et Médical (Industrial, Scientific, and Medical)
- **JMS** : Java Messaging Service
- **JS** : JavaScript
- **LPWAN** : Low Power Wide Area Network (Réseau étendu à faible consommation d'énergie)
- **LWT** : Last Will and Testament (Dernière Volonté et Testament)
- **ML** : Machine Learning (Apprentissage automatique)
- **M2M** : Machine to Machine (Machine à Machine)
- **OASIS** : Organization for the Advancement of Structured Information Standards (Organisation pour l'avancement des normes d'information structurée)
- **PC** : Personal Computer (Ordinateur Personnel)
- **QoS** : Quality of Service (Qualité de Service)
- **RCS** : Réseau de capteurs Sans fil (Wireless Sensor Networks)
  
- **SCADA** : Supervisory Control and Data Acquisition (Contrôle et Acquisition de Données Supervisés)

- **SSL** : Secure Sockets Layer (Couche de Sockets Sécurisés)
- **TCP/IP** : Transmission Control Protocol / Internet Protocol
- **TLS/SSL** : Transport Layer Security / Secure Sockets Layer (Sécurité de la couche de transport / Couche de sockets sécurisés)
- **Wi-Fi** : (Wireless Fidelity)
- **WSN** : Wireless Sensor Networks (Réseaux de capteurs sans fil)
- **XMPP** : Extensible Messaging and Presence Protocol (Protocole de Présence et de Messagerie Extensible)

# Chapitre 01 Introduction Générale

L'Internet des objets (IdO) est un paradigme technologique moderne qui crée un réseau mondial de machines et d'appareils pouvant communiquer entre eux. Les caméras de sécurité, les caméras, les véhicules, les batteries et les logiciels peuvent échanger des données entre eux. L'Internet des objets est reconnu comme l'un des domaines les plus importants pour les technologies futures et est largement étudié lors d'un événement majeur pour les applications et les domaines des maisons et des villes intelligentes pour l'armée, l'éducation, les hôpitaux, les systèmes de sécurité intérieure, les transports et l'agriculture.

Ce paradigme décrit des appareils équipés de capteurs, de puissance de traitement, de logiciels et d'autres technologies qui se connectent et échangent des données avec d'autres appareils et systèmes via internet ou d'autres réseaux de communication. L'Internet des Objets comprend l'électronique, les communications et l'ingénierie informatique.

## 1.1. Contexte

Ce paradigme a évolué grâce à la convergence de nombreuses technologies, notamment l'informatique omniprésente, les capteurs de base, les systèmes embarqués de plus en plus puissants, ainsi que l'apprentissage automatique. Les domaines actuels des systèmes embarqués, des réseaux de capteurs sans fil, des systèmes de contrôle et de l'automatisation (y compris la domotique et l'immotique) permettent de manière indépendante et collective de mettre en œuvre l'Internet des Objets.

Sur le marché grand public, la technologie IdO est synonyme de produits de maison intelligente, y compris les appareils et appareils électroménagers (luminaires, thermostats, systèmes de sécurité domestique, caméras et autres appareils) qui prennent en charge un ou plusieurs écosystèmes courants et peuvent être contrôlés via des appareils. Ainsi, notre destin est entre nos mains, alors pourquoi ne pas le rendre meilleur et plus pratique ?

## 1.2. Objectifs

À l'ère de l'informatique, tout le monde a une vie très chargée. Nous devrions donc penser à vivre confortablement au lieu du stress. Alors, la question est de savoir profiter pleinement du progrès technologique. La domotique est devenue et restera la principale approche pour atteindre cet objectif. Elle est devenue la tendance de notre époque et même pour les années à venir.

La domotique est un terme utilisé pour décrire tous les comforts à l'intérieur d'une maison, et parfois au-delà. Les dispositifs fonctionnent ensemble et devraient être contrôlés facilement depuis nos smartphones et nos tablettes ou des ordinateurs connectés à internet. En conséquence, cela ouvre une porte qui nous relie et nous permet de voyager à travers et vers différents espaces. Ainsi, ces appareils que nous utilisons au quotidien, permettent à chacun de nous de mieux contrôler sa maison.

## 1.3. Méthodologie et résultats

Notre projet a étudié les architectures et plateformes des systèmes IdO, détaillant le protocole MQTT pour la communication entre appareils. Nous avons utilisé des diagrammes UML pour visualiser et planifier l'architecture, facilitant ainsi le développement. Les diagrammes UML ont aidé à comprendre les interactions

entre composants, améliorant performance et scalabilité. L'utilisation de diverses plateformes a offert une flexibilité accrue pour les domaines d'application.

Nous avons développé représente une avancée significative dans le domaine de l'automatisation domestique. Les produits de maison intelligente ont démontré une intégration réussie de la technologie IdO, améliorant confort et sécurité.

## **1.4. Aperçu du rapport**

Nous avons donc décidé de transformer cette idée intéressante en un projet concret. Notre mémoire se compose d'une introduction générale et de quatre chapitres : Dans un premier temps, nous présenterons les architectures et les plateformes des systèmes Internet des Objets. Ensuite, nous détaillerons l'architecture (le protocole) MQTT. Nous décrivons l'implémentation à travers quelques diagrammes UML. Enfin, une conclusion générale sera donnée à la fin de ce mémoire.

# Chapitre 02 Paradigme de IdO dans la vie du quotidien

## 2.1. Introduction

Au cours des dernières décennies, les systèmes domotiques sont devenus de plus en plus populaires, car ils améliorent le confort et la qualité de vie des personnes. L'Internet des Objets (IdO) peut être utilisée pour améliorer tous les aspects de la vie, conduisant à de nombreux scénarios tels que les maisons intelligentes, les voitures autonomes, les villes intelligentes, médecine intelligente ou l'agriculture intelligente. Il s'agit ici, d'intégrer les services de l'Internet des objets (IdO) en injectant l'intelligence dans les capteurs et les actionneurs, en connectant les objets intelligents et en facilitant les interactions entre eux. Ce qui augmente la puissance de calcul, la capacité de stockage et améliore l'efficacité de l'échange de données. De tels systèmes reposent sur la collecte de données collectées pour être utilisées pour suivre, contrôler, traiter et transmettre vos informations à d'autres appareils sur Internet, activer automatiquement certaines actions lorsque certaines situations surviennent. Imaginez maintenant ce qui se passerait si vous n'aviez même pas à entreprendre ces actions. En d'autres termes, votre maison intelligente sait que vous devez effectuer certaines actions et les exécute automatiquement. C'est ce que notre projet tente de faire.

Nous connaissons une abondance de solutions de type middleware dédiées à Internet des Objets. Ces middlewares assurent la connectivité des capteurs et des actionneurs à Internet. Pour être largement adoptées, ces solutions middleware, appelées parfois plateformes, doivent répondre aux attentes des différents acteurs de l'écosystème IdO, notamment les fournisseurs d'appareils, les développeurs d'applications et les utilisateurs finaux.

Dans ce mémoire, nous présenterons un échantillon représentatif de ces plateformes, à la fois propriétaires et open-source, sur la base de leur capacité à répondre aux attentes des différents utilisateurs de l'IdO. L'évaluation se concentre ainsi davantage sur la disponibilité et l'utilisabilité de ces plateformes pour les acteurs de l'écosystème IdO, plutôt que sur les particularités des couches technologiques sous-jacentes. L'évaluation est réalisée sous la forme d'une analyse des lacunes du paysage actuel de l'IdO en ce qui concerne (i) la prise en charge de technologies de détection et d'actionnement hétérogènes, (ii) la propriété des données et ses implications pour la sécurité et la confidentialité, (iii) le traitement des données et l'utilisation des données. Les capacités de partage, (iv) le support offert aux développeurs d'applications, (v) l'exhaustivité d'un écosystème IdO, et (vi) la disponibilité de places de marché IdO dédiées.

Ces écarts visent à mettre en évidence les déficiences des solutions d'aujourd'hui pour améliorer leur intégration aux écosystèmes de demain. Des auteurs ont présenté les résultats d'une analyse suite à une enquête menée auprès des partenaires du programme finlandais IdO, comptant plus de 350 experts, afin d'évaluer les problèmes les plus critiques pour le développement des futures plateformes IdO. Sur la base des résultats de cette analyse, les auteurs ont présenté une liste de recommandations pour étendre ces plateformes IdO afin de combler les lacunes. Le paradigme de l'Internet des Objets prévoit le développement de notre environnement actuel vers de nouveaux espaces enrichis, tels que les villes intelligentes, les maisons intelligentes, les réseaux intelligents, la santé numérique et le contrôle automatisé de la pollution environnementale [20-21-22].

Ces dernières années, de nombreuses solutions ont émergé pour interconnecter des objets intelligents pour des systèmes ayant des échelles et des objectifs différents. Par exemple, une plateforme légère peut être déployée chez-soi pour orchestrer plusieurs objets connectés, comme le réfrigérateur, les lumières et le système de chauffage. À une échelle plus large, une ville intelligente peut bénéficier de son développement et



de sa gestion grâce à de nouvelles solutions IdO capables de gérer des milliers de capteurs, de faciliter leur maintenance, leur recalibrage et, plus important encore, d'analyser les données qu'ils produisent.

Dans ce qui suit, nous présenterons le paysage IdO actuel en ce qui concerne la distribution d'applications et de services, ainsi que les plateformes qui connectent les appareils à Internet. Pour les besoins de cet article, une plate-forme IdO est définie comme le middleware et l'infrastructure qui permettent aux utilisateurs finaux d'interagir avec des objets intelligents, comme le montre la figure 1. L'objectif de notre analyse de ces papiers est de présenter les lacunes de ces plates-formes en ce qui concerne leurs capacités à relever les défis émergeant du développement actuel des technologies IdO. Afin d'évaluer les limites du paysage actuel des plateformes IdO et d'identifier les lacunes qui doivent être comblées, nous prenons en compte les points de vue de différents acteurs de l'écosystème des plateformes IdO, notamment les fournisseurs d'appareils, les développeurs d'applications, les fournisseurs de plateformes et de services associés, et les utilisateurs finaux.

## 2.2. Architecture

L'Internet des objets est un concept qui décrit la connexion de divers appareils dotés de capteurs et d'équipements de communication intégrés pour réaliser la collecte et la transmission de données dans un réseau. Les appareils IdO se multiplient avec des progrès géométriques, et garantir l'interopérabilité et la gestion des énormes données hétérogènes générées est d'une importance majeure pour la création d'applications et de services intelligents.

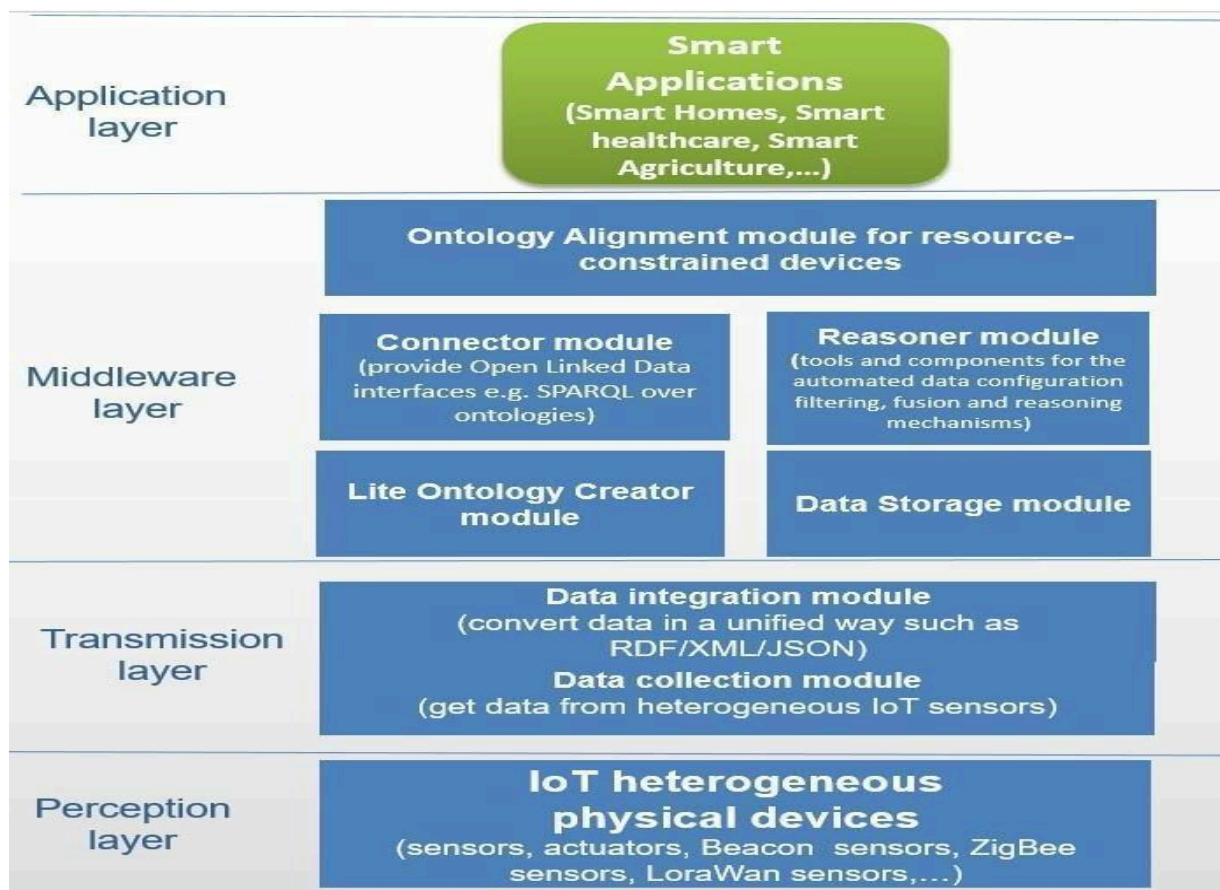


Figure 1 Architecture IdO [20]

## 2.3. Plateforme Arduino

Arduino est une plate-forme électronique open source basée sur du matériel et des logiciels faciles à utiliser. Les cartes Arduino sont capables de lire des entrées - une lumière sur un capteur, un doigt sur un bouton ou un message Twitter - et de les transformer en sortie - en activant un moteur, en allumant une LED, en publiant quelque chose en ligne. Vous pouvez indiquer à votre carte quoi faire en envoyant un ensemble d'instructions au microcontrôleur de la carte. Pour ce faire, il est impératif d'utiliser le langage de programmation Arduino (basé sur le câblage) et le logiciel Arduino (IDE), basé sur le traitement. Voici un aperçu de ce qu'est Arduino et de son fonctionnement [22].

### 2.3.1. Arduino Fonctionnement

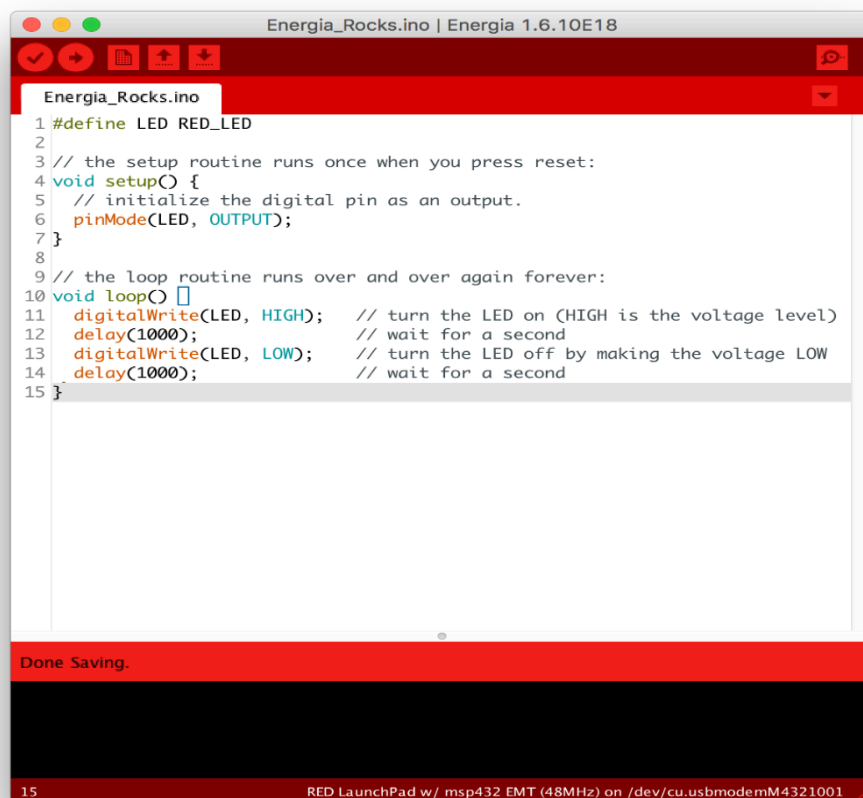
- **Matériel** : Les cartes Arduino sont des microcontrôleurs programmables dotés de divers ports d'entrée/sortie (E/S) qui permettent de connecter des capteurs, des actionneurs et d'autres composants électroniques. Les cartes Arduino sont disponibles dans différentes versions, chacune offrant des fonctionnalités spécifiques et des configurations matérielles différentes.
- **Logiciel** : Arduino est associé à un environnement de développement intégré (IDE) qui permet de programmer les cartes Arduino. L'IDE Arduino est basée sur le langage de programmation Wiring, simplifié pour les débutants, mais assez puissant pour des applications avancées. Il comprend un éditeur de code, un compilateur, un gestionnaire de bibliothèques et des outils de téléchargement.
- **Programmation** : Les programmes Arduino, appelés "sketches", sont écrits en langage C/C++. Ils sont chargés sur la carte Arduino via un câble USB depuis l'IDE Arduino. Les sketches peuvent contrôler les entrées et sorties de la carte Arduino, lire les capteurs, activer les actionneurs, effectuer des calculs, communiquer avec d'autres périphériques, etc.
- **Projets** : Arduino est utilisé dans une grande variété de projets électroniques, allant des simples clignotements de LED aux systèmes domotiques complexes, en passant par les robots, les drones, les dispositifs médicaux, les instruments de mesure, les œuvres d'art interactives, et bien plus encore. Sa simplicité d'utilisation et sa flexibilité en font un choix populaire pour les makers, les étudiants, les amateurs et les professionnels.
- **Communauté** : Arduino bénéficie d'une vaste communauté d'utilisateurs et de développeurs qui partagent leurs projets, leurs connaissances et leur soutien sur des forums en ligne, des tutoriels, des livres et des événements locaux et internationaux. Cette communauté dynamique contribue à faire d'Arduino une plateforme accessible et collaborative pour l'innovation et la créativité.

Arduino est bien plus qu'un simple outil de développement électronique ; c'est une plateforme versatile et communautaire qui encourage l'expérimentation, l'apprentissage et l'innovation dans le domaine de l'électronique et de la programmation [22].

## 2.3.2 Energia

Energia est une plate-forme de prototypage électronique open source lancée par Robert Wessels en janvier 2012 dans le but d'intégrer le framework Wiring et Arduino au LaunchPad basé sur Texas Instruments MSP430. L'IDE Energia est multiplate-forme et prise en charge sur Mac OS, Windows et Linux. Energia utilise le compilateur mspgcc de Peter Bigot et est basé sur le framework Wiring et Arduino. Energia comprend un environnement de développement intégré (IDE) qui repose sur l'IDE de traitement (Processing→Wiring→Arduino→Energia). Elle est également un Framework/couche d'abstraction portable qui peut être utilisé dans d'autres IDE populaires. Utilisez un environnement basé sur un navigateur Web avec Texas Instruments CCS Cloud sur dev.ti.com ou le puissant IDE CCS Desktop de TI.

La base d'Energia et d'Arduino est le Framework Wiring développé par Hernando Barragan. Le cadre est soigneusement créé en pensant aux designers et aux artistes pour encourager une communauté où les débutants et les experts du monde entier partagent des idées, des connaissances et leur expérience collective. L'équipe Energia adopte la philosophie de l'apprentissage par la pratique et s'efforce de faciliter le travail directement avec le matériel. Les ingénieurs professionnels, les entrepreneurs, les créateurs et les étudiants peuvent tous bénéficier de la facilité d'utilisation qu'Energia apporte au microcontrôleur. En collaboration avec Energia, LaunchPad peut être utilisé pour développer des objets interactifs, en prenant les entrées d'une variété de commutateurs ou de capteurs et en contrôlant une variété de lumières, de moteurs et d'autres sorties physiques. Les projets LaunchPad peuvent être autonomes (exécutés uniquement sur la carte cible, c'est-à-dire votre LaunchPad), ou ils peuvent communiquer avec un logiciel exécuté sur un ordinateur (PC hôte) [22].



```
1 #define LED RED_LED
2
3 // the setup routine runs once when you press reset:
4 void setup() {
5   // initialize the digital pin as an output.
6   pinMode(LED, OUTPUT);
7 }
8
9 // the loop routine runs over and over again forever:
10 void loop() {
11   digitalWrite(LED, HIGH); // turn the LED on (HIGH is the voltage level)
12   delay(1000); // wait for a second
13   digitalWrite(LED, LOW); // turn the LED off by making the voltage LOW
14   delay(1000); // wait for a second
15 }
```

Figure 2 code arduino [22].

## 2.4. Maison intelligente dans l’histoire et le futur

Selon l'étude des auteurs [4], il n'a pas fallu longtemps pour que les inventeurs et les fabricants reconnaissent le potentiel d'électricité à usage domestique. Au milieu des années 1900, les maisons possédaient un large éventail de caractéristiques qui étaient inimaginables la décennie précédente. Chanceux pour nous tous, les appareils domestiques sont depuis devenus plus petits, plus élégants et plus avancés. Aujourd'hui, disposer d'un assistant virtuel tel qu'Alexa ou Google Home est la norme pour les familles ordinaires. Ces appareils permettent de régler des minuteries et d'écouter de la musique. La sécurité à domicile n'a jamais non plus été plus évoluée qu'elle ne l'est actuellement. Les vidéos vous permettent de vérifier qui se trouve devant votre porte en transmettant des images en temps réel sur votre téléphone. Vous pouvez même contrôler vos serrures de porte, vos portes de garage et votre caméra depuis votre téléphone ou votre tablette. Des robinets intelligents dotés de capteurs de mouvement aident à économiser l'eau et font de la maison un centre de connectivité en constante évolution.

La domotique, quant à elle, consiste à connecter tous vos appareils à Internet. Imaginez pouvoir tout contrôler dans votre maison avec juste une touche ou via une reconnaissance vocale. Il n'y a pas si longtemps, il fallait courir vers la radio pour appuyer sur les touches « play » et « record » en même temps pour enregistrer votre chanson préférée sur un lecteur de cassettes. Comme des dispositifs continuent de progresser, les utilisateurs peuvent s'attendre à d'importantes économies d'argent et d'énergie. Les usagers peuvent s'attendre également à ce que la technologie anticipe leurs besoins et leurs désirs avant d'en être informés. Le développement d'appareils et de sécurité plus intelligents peut également être anticipé. Des lumières qui s'allument et s'éteignent tout seul sans être invité à ouvrir des portes avec une reconnaissance faciale. Les possibilités sont apparemment infinies à mesure qu'il continue d'évoluer et de gagner en popularité, les utilisateurs peuvent s'attendre à économiser du temps et de l'argent.

## 2.5. Types de systèmes de maison intelligente

Il existe deux types de systèmes de maison intelligente.

- Systèmes sans fil (Wi-Fi, radio, etc.)

Le Wi-Fi occupe une position unique pour prendre en charge les applications IdO qui nécessitent une connectivité Internet pour fonctionner. Certaines applications, telles que les services automobiles et les applications vidéo, telles que les caméras de surveillance du réseau, nécessitent la bande passante d'un réseau haut débit sans fil pour répondre à d'autres exigences [24] [25].

- Système de câble intelligent (système filaire).

Le câblage intelligent fait partie d'une maison intelligente et contribue à la domotique. De nombreux appareils domestiques intelligents sont sans fil, mais d'autres nécessitent une connexion physique directe. Tous les appareils contrôlés par une maison intelligente ne sont pas eux-mêmes intelligents. Ce sont les combinaisons créées entre eux qui rendent le système intelligent, en lui permettant d'utiliser différents types de câbles circulant à l'intérieur de la maison. Avec un seul emplacement, vous pouvez facilement contrôler l'intégration de divers systèmes et capteurs de maison intelligente dans toute votre maison.

## 2.6. Maisons intelligentes en Algérie (Statistiques)

Premièrement, le nombre d'entreprises opérant dans notre pays est faible par rapport à celles opérant réellement à l'étranger. Cela est dû à l'absence de grandes entreprises de ce secteur qui opèrent volontairement ou involontairement sur le marché intérieur, ou à l'exposition de technologies qui ne sont pas adaptées au style et au niveau de vie des Algériens. La domotique est rarement vue ou peu utilisée dans les ménages algériens. C'est parce que la nécessité et l'inévitabilité ne sont pas perçues.

Dans un pays où les immeubles avec ascenseurs sont considérés comme du luxe/prestige, les appartements, livrés en 2001, sont toujours inachevés et toujours privés d'électricité et d'eau. En Algérie, les protocoles de contrôle électronique à distance (éclairage automatique, irrigation, ouverture et fermeture automatiques des portes, garages et fenêtres) sont rarement utilisés, même si les systèmes d'énergie électrique répondent aux besoins locaux.

Cependant, la domotique en Algérie a longtemps été défavorable en raison de conditions de faible productivité. L'utilisation des paiements en ligne est faible, la surveillance à distance n'est pas facile et l'installation de caméras numériques est très complexe et prend beaucoup de temps. Des incidents domestiques potentiellement dangereux tels que fuites d'eau, fuites de gaz, incendies, inondations et autres problèmes qui doivent être résolus. Par conséquent, nous pensons que ce projet est suffisamment important pour y investir du temps et des efforts. Nous visons un avenir meilleur et, espérons-le, améliorerons notre niveau de vie dans les années à venir [41] [42].

## 2.7. Conclusion

L'Internet des Objets représente une révolution technologique majeure avec un impact profond sur notre façon de vivre, de travailler et d'interagir avec le monde qui nous entoure. Cette convergence d'objets physiques, de capteurs, de logiciels et de connectivité Internet ouvre de nouvelles possibilités infinies, mais elle présente également des défis significatifs.

Compte tenu notamment de la situation des maisons intelligentes en Algérie, notre projet résout les problèmes fondamentaux que nous pouvons rencontrer et, bien que sa mise en œuvre soit en attente, il existe des moyens plus efficaces pour faciliter la vie. L'objectif est de montrer qu'un tel projet est viable et hautement réalisable en trouvant un moyen de le faire.

# Chapitre 03 Architecture MQTT

## 3.1. Introduction

Dans le domaine de la communication pour les maisons intelligentes, nous trouvons deux objectifs ambitieux. Le premier est de savoir comment rendre possible la communication des équipements à l'intérieur de la maison. Le deuxième est de connecter la maison intelligente au monde extérieur. Cependant, la partie la plus importante dépend du choix de la structure de connexion appropriée, y compris le protocole de messagerie efficace. Par conséquent, avant de sélectionner le protocole approprié pour les systèmes IdO, le prérequis est de mieux comprendre un système IdO cible et ses exigences en matière de partage de messages/données.

Contrairement au Web, qui utilise un protocole de messagerie standard unique (HTTP/HTTPS), l'IdO ne peut pas s'appuyer sur un protocole unique pour tous ses besoins. Ce que nous entendons par là, c'est que dans la structure IdO, nous pouvons utiliser plusieurs protocoles, où chacun effectue une tâche spécifique, par rapport à un protocole unique conçu uniquement pour un seul but. Prenons l'exemple du protocole Http qui a été implémenté dans le but d'accéder à des pages Web ou sites Internet. Par conséquent, de nombreux protocoles de messagerie sont disponibles pour différents types d'exigences du système IdO. Certains d'entre eux ont été conçus pour répondre à des applications nécessitant des transactions commerciales rapides et fiables telles que AMQP (Advanced Message Queuing Protocol) et JMS (Java Message Service). Ainsi, de nombreux protocoles ont été conçus pour répondre aux applications nécessitant la collecte de données dans des réseaux contraints tels que MQTT et CoAP. Beaucoup d'entre eux ont été conçus pour répondre aux applications nécessitant une messagerie instantanée et une détection de présence en ligne telles que XMPP et SIP. Quelques-uns d'entre eux ont été conçus pour répondre aux applications Web nécessitant de communiquer via l'internet, tels que les protocoles client/serveur Restful HTTP et CoAP [4].

Cela montre clairement que l'avenir de l'IdO réside dans plusieurs protocoles de messagerie, mais cela dépend aussi principalement du cas étudié ou utilisé, par exemple, dans notre cas, nous nous sommes concentrés principalement sur un protocole, qui est responsable de la communication des appareils. Il est donc nécessaire d'examiner les avantages et les inconvénients des protocoles de messagerie largement acceptés et émergents pour les systèmes IdO afin de déterminer leurs scénarios les plus adaptés.

## 3.2. Couche des dispositifs IdO

### 3.2.1. Les dispositifs d'une maison intelligente

Dans le contexte d'une maison intelligente, la couche de dispositif IdO joue un rôle crucial en assurant l'interaction avec l'environnement physique. Cette couche comprend divers capteurs, actionneurs, microcontrôleurs, et modules de communication qui permettent aux dispositifs de collecter des données, de communiquer entre eux et de répondre aux commandes des utilisateurs. Voici un aperçu détaillé de cette couche dans une maison intelligente.

#### 3.2.1.1. Composants Principaux de la Couche de Dispositif IdO

a) *Les capteurs :*

**Capteurs de Mouvement :** Détectent la présence ou le mouvement dans une pièce pour activer les lumières ou les systèmes de sécurité.

**Capteurs de Température et d'Humidité :** Mesurent les conditions environnementales pour réguler le chauffage, la climatisation ou la ventilation.

**Capteurs de Lumière** : Mesurent le niveau de luminosité pour ajuster automatiquement les stores ou l'éclairage.

**Capteurs de gaz et Monoxyde de Carbone** : Détectent la présence de fumée ou de gaz dangereux pour avertir les résidents et déclencher des alarmes.

*b) Actionneurs :*

**Interrupteurs et Déclencheurs Intelligents** : Contrôlent l'allumage, l'extinction et l'intensité des lumières.

**Serrures Intelligentes** : Verrouillent et déverrouillent les portes à distance.

**Thermostats Intelligents** : Régulent la température intérieure en fonction des préférences et des conditions météorologiques.

*c) Microcontrôleurs et microprocesseurs :*

**Arduino** : Utilisé pour des projets de bricolage et des systèmes de capteurs simples.

**ESP8266/ESP32** : Fournit des capacités de communication Wi-Fi pour des dispositifs connectés à Internet.

**Raspberry Pi** : Utilisé pour des applications plus complexes nécessitant une puissance de traitement supérieure et des capacités de communication variées.

*d) Modules de Communication :*

**Wi-Fi** : Permet la communication sans fil avec le réseau domestique et les serveurs cloud.

**Bluetooth** : Utilisé pour la communication à courte portée entre les appareils, souvent utilisé pour les appareils portables et les contrôleurs.

**Zigbee et Z-Wave** : Protocoles de communication sans fil à faible consommation d'énergie, couramment utilisés pour les dispositifs de maison intelligente.

**LoRa** : Utilisé pour la communication longue portée à faible consommation d'énergie, souvent dans des environnements avec de nombreux dispositifs répartis sur une large zone.

### 3.2.2. Considérations pour la Conception de la Couche de Dispositif IdO

- **Consommation d'Énergie** : Les dispositifs doivent être conçus pour minimiser la consommation d'énergie, surtout ceux fonctionnant sur batterie, afin de prolonger leur durée de vie opérationnelle.
- **Fiabilité et Robustesse** : Les dispositifs doivent être fiables et capables de fonctionner dans des conditions variées, y compris les variations de température, d'humidité et d'autres facteurs environnementaux.
- **Sécurité** : Les dispositifs IdO doivent intégrer des mesures de sécurité robustes pour protéger contre les accès non autorisés et les cyberattaques.
- **Interopérabilité** : Les dispositifs doivent pouvoir communiquer et interagir avec d'autres dispositifs et plateformes de différentes marques et technologies.
- **Facilité d'Installation et d'Utilisation** : Les dispositifs doivent être faciles à installer et à utiliser, avec des interfaces utilisateur intuitives et une documentation claire.
- **Capabilité** : La capacité à ajouter facilement de nouveaux dispositifs ou à étendre le système est essentielle pour permettre la croissance et l'évolution de la maison intelligente.

### 3.2.3. Le scénario

Les figures de 3-12 ci-dessous présentent des prises de vues de notre application pour le contrôle d'une maison intelligente et utilisées pour décrire quelques scénarios pour un système dédié au contrôle d'une maison intelligente.

- **Éclairage automatique sur la maison et la surveillance** : utilisez des capteurs de mouvement et de lumière pour ajuster automatiquement l'éclairage en fonction de la présence et des conditions d'éclairage.

Ceci dans le but de réduire la consommation et d'organiser l'éclairage à l'intérieur de la maison pour améliorer la vie des propriétaires, en particulier des enfants et de leurs belles-mères.

Il fonctionne automatiquement lorsque le niveau d'éclairage est faible et éclaire le lieu à une vitesse déterminée par l'utilisateur. De plus, lorsque des détecteurs de mouvement détectent un mouvement dans le lieu, il allume les lumières et, dans des endroits spécifiques comme l'entrée de la maison et le garage, et il active les caméras pour surveiller et connaître l'identité de l'individu.

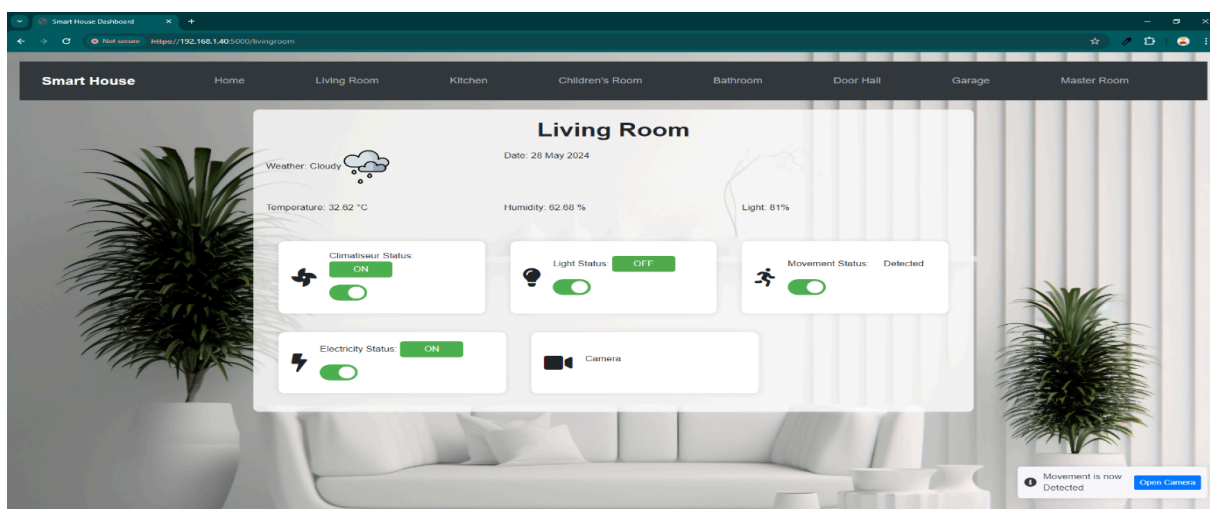


Figure 3 Vue générale de notre application pour contrôler une maison.

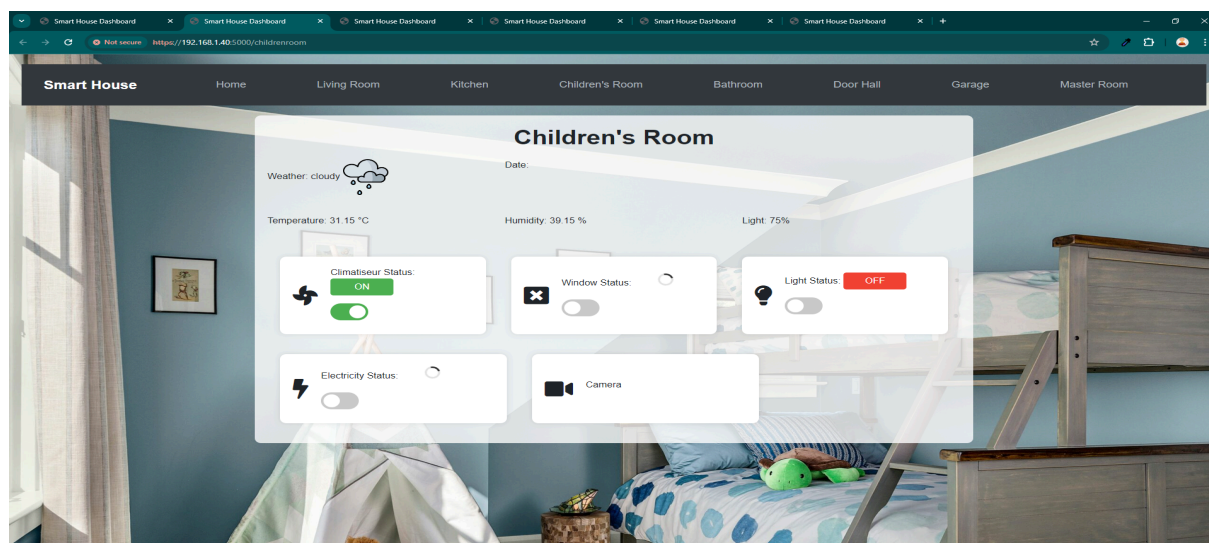


Figure 4 capture sur le pc (Dashboard) le salon et la chambre d'enfant.



- **Contrôlée la Climatisation** : les thermostats intelligents et les capteurs de température intégrés optimisent la consommation d'énergie et maintiennent un confort optimal à l'intérieur des pièces en régulant la température ambiante dans une certaine mesure. Lorsque la température augmente, le climatiseur passe en mode refroidissement. Au fur et à mesure que la température diminue, le mode chaud s'active selon le désir et le confort de l'utilisateur.

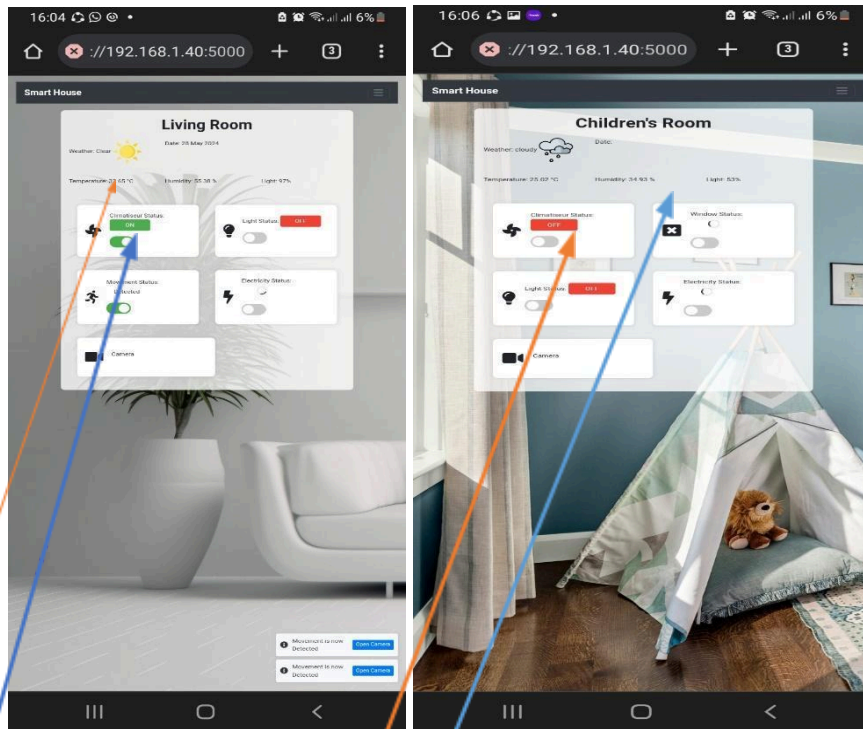


Figure 5 capture sur le smart phone le salon et la chambre d'enfant.

-La climatisation déclenchée.

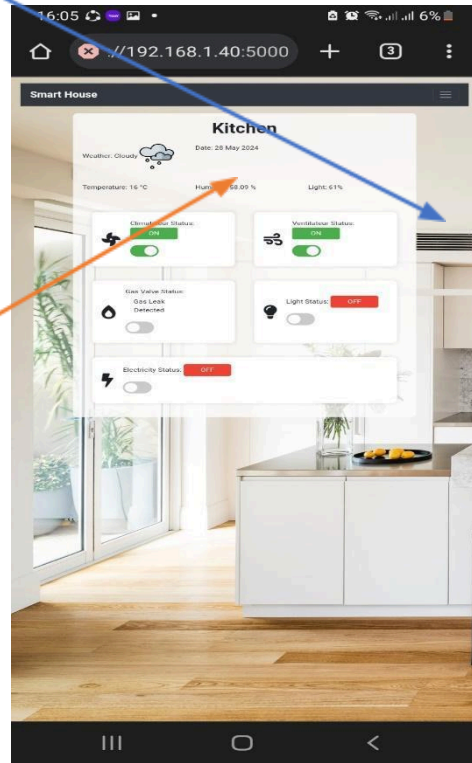
-La température augmentée de plus de 27°.

-La température moins de 27°

-La climatisation quant à elle a reste sur le mode OFF

- **Contrôler l'humidité:** pratique dans les maisons intelligentes, notamment dans les zones comme la cuisine et la salle de bains où les niveaux d'humidité peuvent fluctuer considérablement. Intégrer des capteurs d'humidité dans ces pièces permet de surveiller et de réguler l'humidité pour prévenir les problèmes tels que la moisissure, les odeurs désagréables et les dommages matériels. Voici comment cela peut être mis en œuvre

**- La ventilation déclenchée**



**-L'humidité augmente de plus de 50%**

Figure 6 capture la cuisine dans le smart phone.

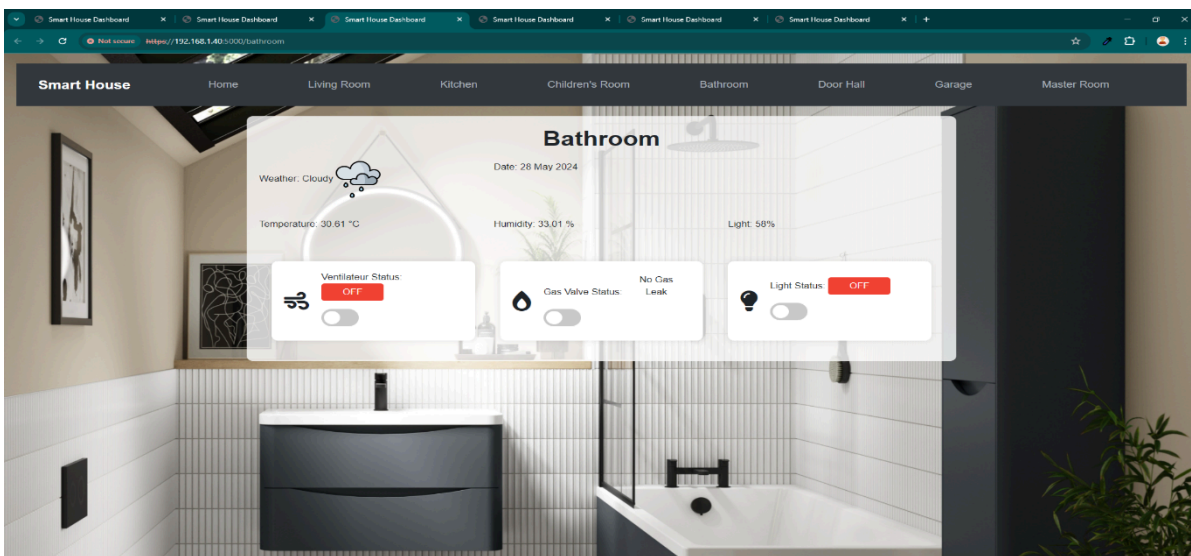


Figure 7 capture sur le pc (Dashboard) la salle de bain

- **Composants et Fonctionnement**

### Capteurs d'Humidité

Fonction : Mesurent les niveaux d'humidité relative dans l'air.

Placement : Installés dans la cuisine et la salle de bains à des endroits stratégiques (proche des éviers, des douches, des cuisinières).

### Actionneurs

Ventilateurs d'Extraction : Activés automatiquement ou à distance pour réduire l'humidité lorsque celle-ci dépasse un seuil prédéfini.

Déshumidificateurs : Activés pour absorber l'excès d'humidité.

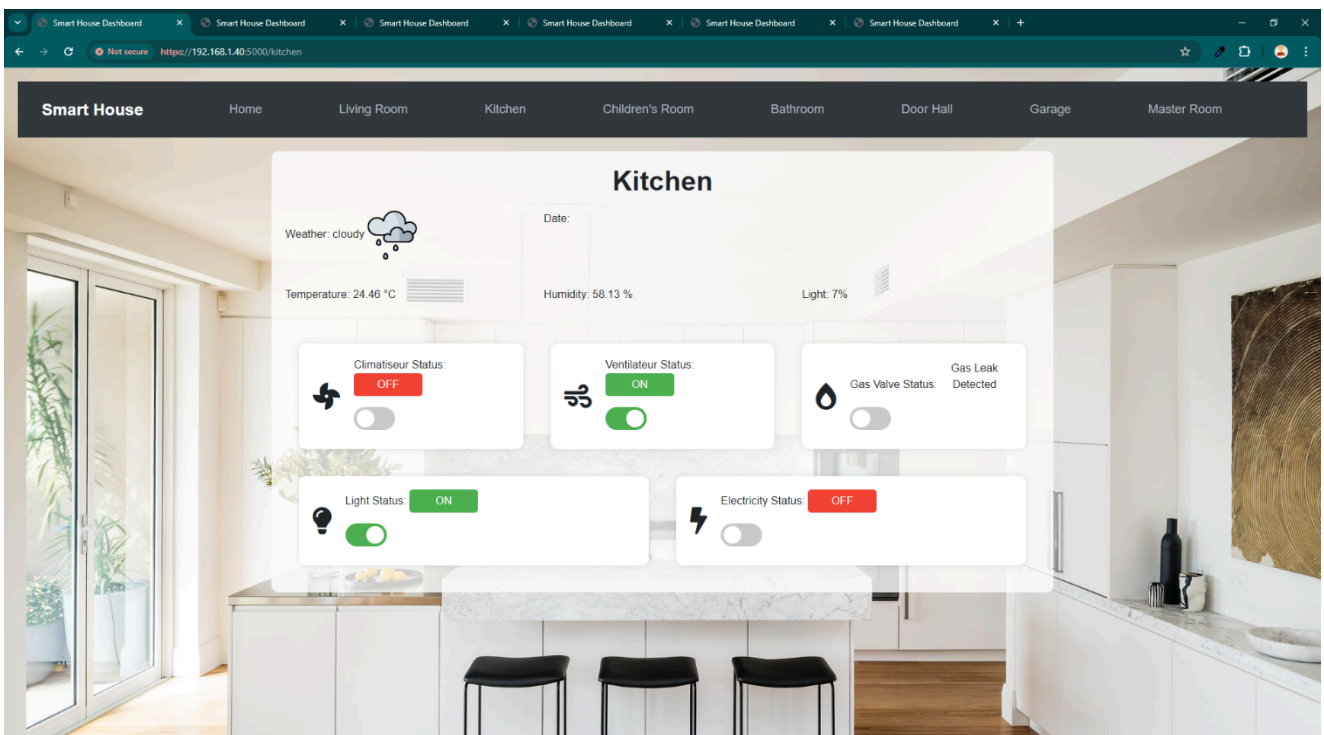


Figure 8 capture sur le pc (Dashboard) de la cuisine.

- **La sécurité domestique :** En intégrant des caméras de sécurité, des capteurs de porte/fenêtre et des alarmes connectées, les résidents peuvent surveiller et protéger leur domicile de manière plus efficace et pratique. Voici une vue d'ensemble de la façon dont ces dispositifs peuvent être intégrés pour une surveillance et une protection accrues :

### Composants de la Sécurité Domestique

#### 1. Caméras de Sécurité

- Fonction : Capturent des vidéos en temps réel des zones surveillées, offrant une surveillance visuelle
- Placement : Entrées principales, arrière-cour, garage, intérieur de la maison.

## Capteurs de Porte/Fenêtre

- Fonction : Détectent l'ouverture et la fermeture des portes et fenêtres, alertent les résidents en cas d'intrusion potentielle.
- Placement : Toutes les portes et fenêtres accessibles depuis l'extérieur.

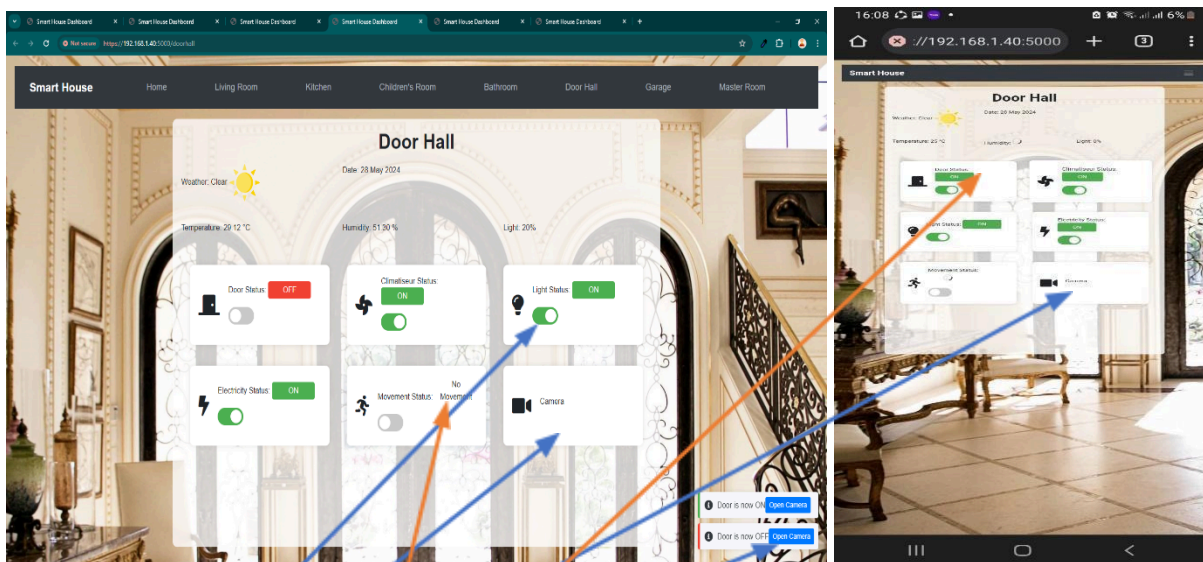


Figure 9 capture sur le pc (Dashboard) du Hall et dans le smart phone.

**-Si le capteur de mouvement ou bien le capteur porte détecte l'ouverture.**

**-Allumé la lumière et envoyer des notifications d'alerte sur le smart phone avec la vue directe grâce à des caméras de surveillance.**

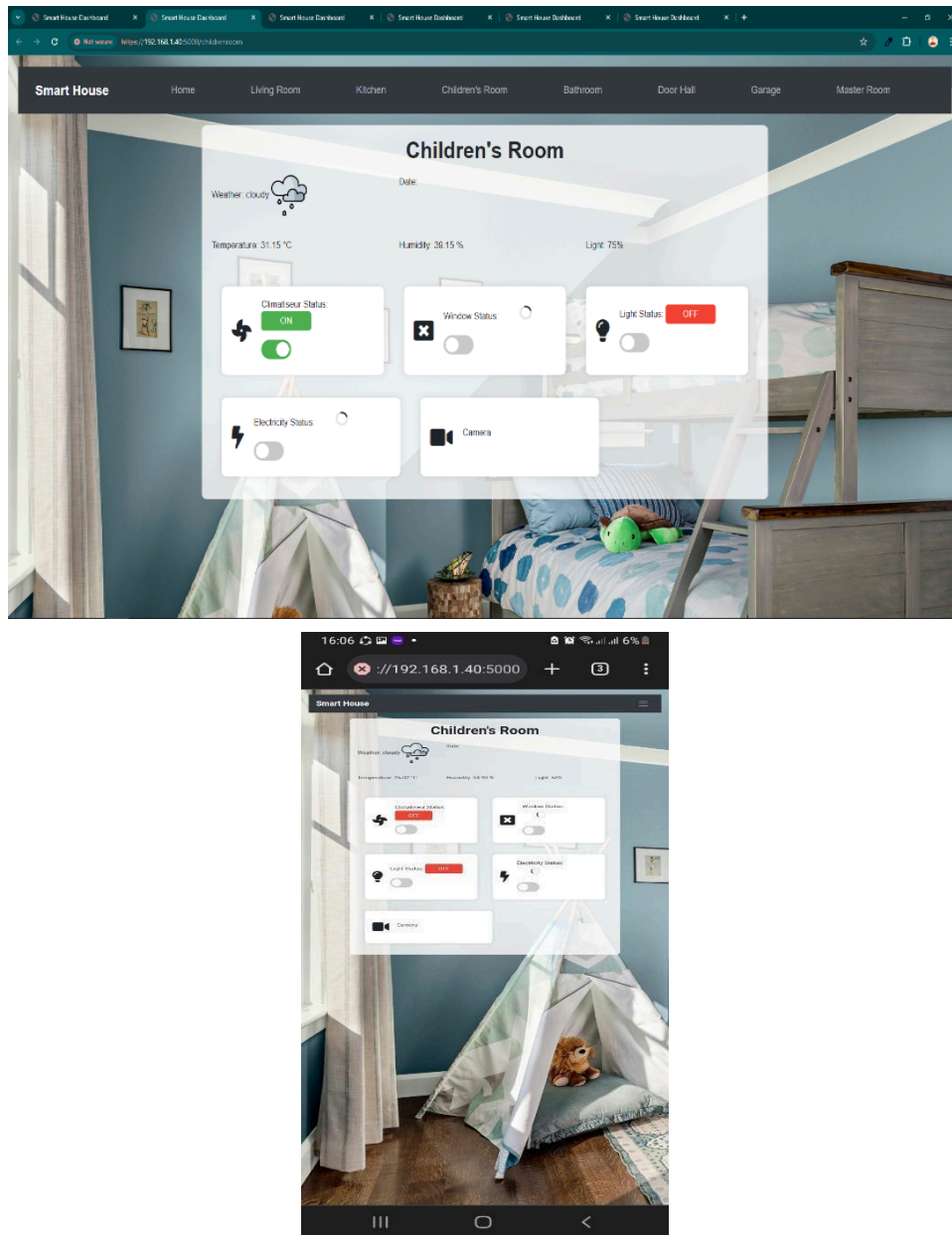


Figure 10 capture sur le pc (Dashboard) de la chambre d'enfant et dans le smart phone.

## 2. Capteurs de Gaz

- Fonction : Détectent la présence de gaz dangereux (comme le gaz naturel, le propane, le monoxyde de carbone).
- Placement : Installés près des sources potentielles de fuite (cuisinières, chaudières – garage, chauffe-eau, etc.).

## 3. Alarmes Connectées

- Fonction : Émettent des alertes sonores et envoient des notifications aux résidents et aux services de sécurité en cas d'intrusion ou d'autre incident détecté.
- Placement : Points d'entrée principaux, endroits stratégiques à l'intérieur de la maison.

- Fonction d'alerte gaz : Avertir les résidents en cas de fuite de gaz via des alarmes sonores et des notifications sur les smartphones.

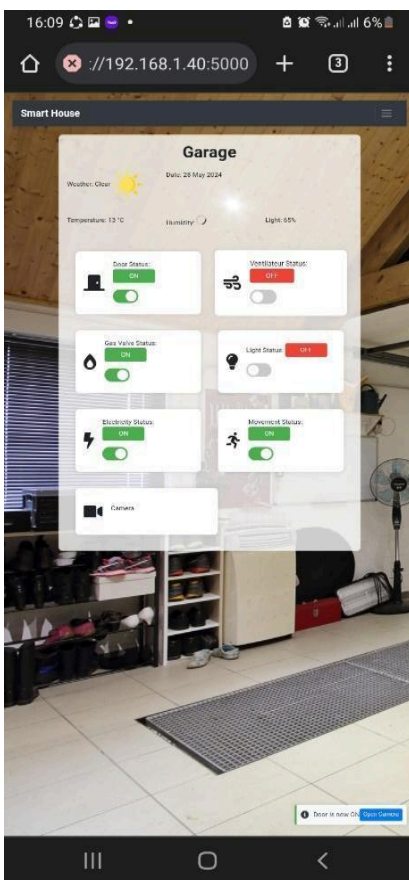
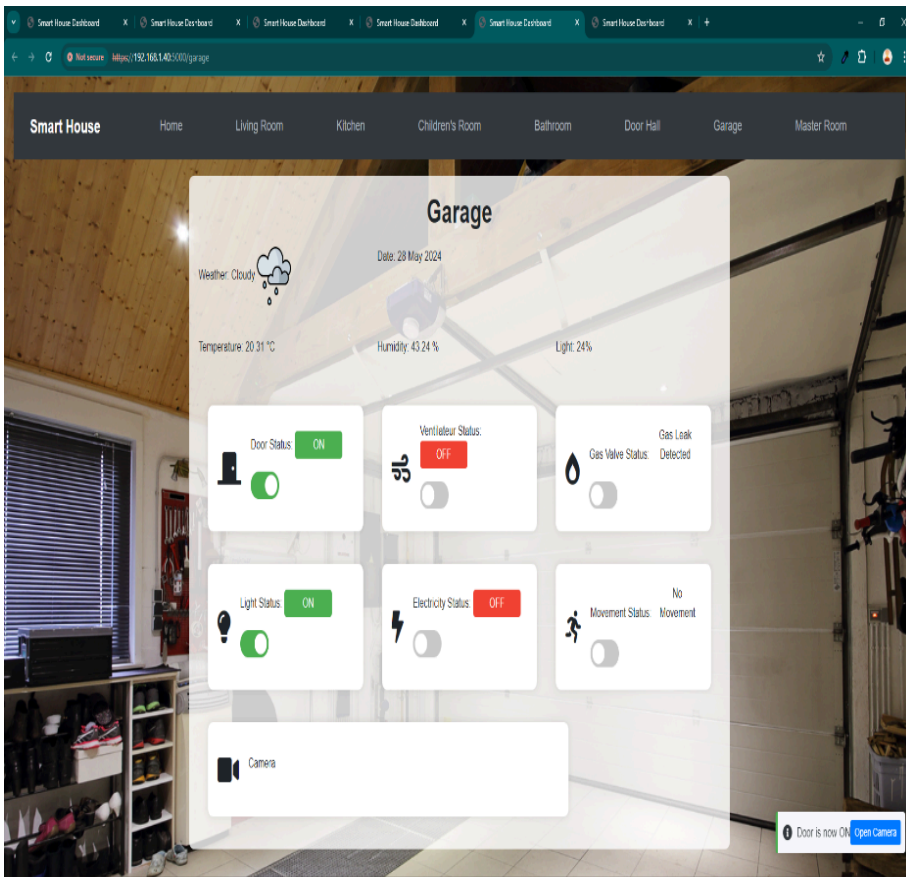


Figure 11 capture sur le pc (Dashboard) du garage et dans le smart phone.

- **Si les capteurs des gaz et de CO détectent le gaz ou une augmentation de CO**
- **Directement alerté et l'alarme sur la maison et envoyer une notification de danger sur le smart phone, ferme la vanne de gaz, couper l'électricité et déclencher les ventilateurs de secoure.**

## **Intégration et Fonctionnement**

### **1. Surveillance Continue**

- **Caméras de Sécurité** : Capturent des images et des vidéos en temps réel, les transmettant au système central ou au cloud pour l'accès à distance.
- **Capteurs de Porte/Fenêtre** : Surveillent l'état des portes et fenêtres et envoient des alertes en cas d'ouverture non autorisée.
- Les capteurs de gaz mesurent en continu les niveaux de gaz dans l'air.

### **2. Détection et Notification**

- **Détection de Mouvement** : Les caméras de sécurité et les capteurs détectent les mouvements ou les ouvertures et envoient des notifications instantanées aux résidents via une application mobile.
- **Activation des Alarmes** : Les alarmes connectées se déclenchent pour alerter les résidents et potentiellement effrayer les intrus.
- Lorsque la concentration de gaz dépasse un seuil prédéfini, les capteurs envoient des données aux microcontrôleurs.
- Ils peuvent vérifier l'état des capteurs et fermer les vannes de gaz via une application mobile.



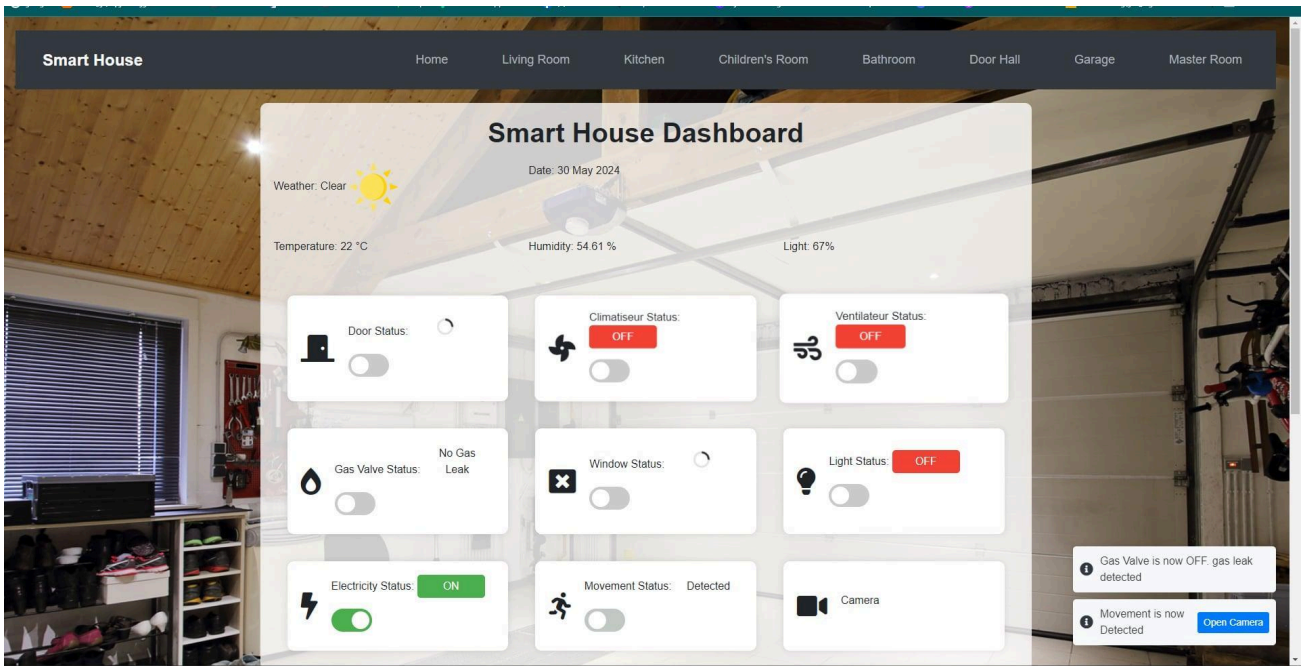


Figure 12 capture sur le pc (Dashboard) de Smart home.

### 3.3. Couche de communication

#### 3.3.1. LoRaWAN (long-range wide area network)

LoRaWAN (LoRa Wide Area Network) sont des technologies de communication sans fil spécialement conçues pour répondre aux besoins de l'Internet des Objets (IdO). Elles se caractérisent par une consommation d'énergie très faible et une grande portée, permettant ainsi de connecter des dispositifs sur de longues distances.

Le protocole de modulation utilisé dans les réseaux LPWAN (Low Power Wide Area Network) fonctionne sur des bandes de fréquences ISM (Industriel, Scientifique, and Médical) sans licence, notamment :

- 868 MHz en Europe
- 915 MHz en USA
- 433 MHz en Asie

##### 3.3.1.1. Caractéristique de LoRaWAN

**1-Modulation** : Utilise la modulation par étalement de spectre (Chirp Spread Spectrum, (CSS) pour assurer une communication robuste et à longue portée.

**2-Portée** : Jusqu'à 15 km en milieu rural et 2 à 5 km en milieu urbain.

**3-Consommation d'énergie** : Très basse consommation, idéale pour les dispositifs IdO alimentés par batterie.

[2]

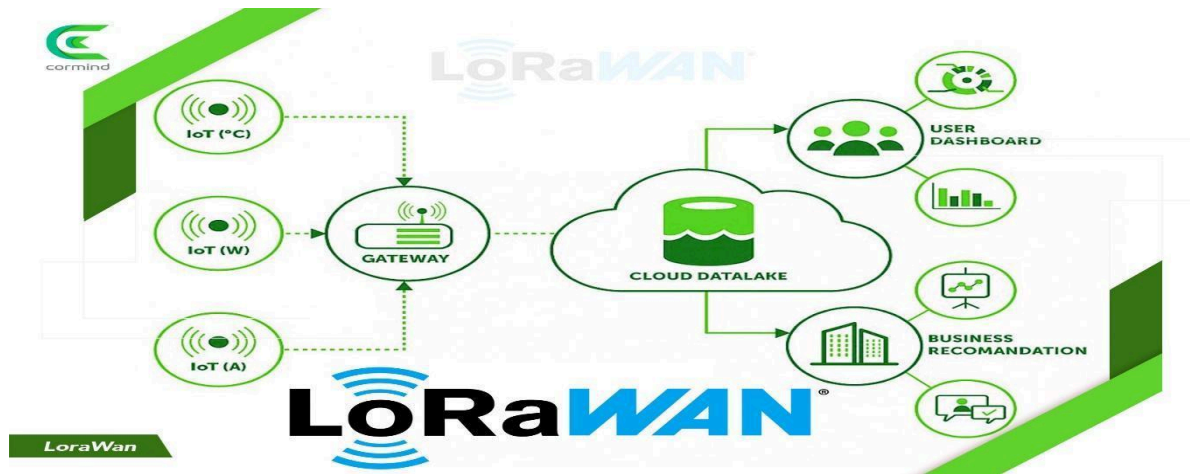


Figure 13LoRaWAN [5]

### 3.3.1.2. Architecture de LoRaWAN le protocole réseau

LoRaWAN est la couche réseau et protocolaire qui fonctionne au-dessus de Lora. Elle définit la communication entre les dispositifs IdO.

#### 1-Dispositifs IdO

- Capteurs et actionneurs équipés de modules Lora.
- Envoyent des données à des intervalles réguliers ou en réponse à des événements.

#### 2-Passerelles

- Recevront les messages des dispositifs IdO et les transmettent aux serveurs via le réseau.
- Fonctionnent comme des ponts entre les dispositifs IdO et l'infrastructure réseau.

#### 3-Serveurs de Réseau

- Gèrent les communications entre les passerelles et les serveurs d'application.
- Effectuent la déduplication des messages, la gestion de la sécurité et la répartition des charges.

#### 4-Serveurs d'Application

- Hébergent les applications IdO qui utilisent les données collectées.
- Peuvent être intégrés avec des systèmes d'analyse de données, des tableaux de bord, et d'autres outils d'exploitation. [1]



Figure 14 LoraWAN [6]

### 3.3.2. Sigfox

Sigfox est un opérateur de télécommunications français, fondé en 2009 par Christophe Fourtet et Ludovic Le Moan, spécialisé dans les services de connectivité pour l'Internet des Objets. Basée à Labège, près de Toulouse, Sigfox propose un réseau cellulaire dédié à l'IdO, comptant 2 600 antennes en France fin 2020. En juin 2021, l'opérateur a annoncé son intention de s'étendre en Afrique et au Moyen-Orient. [30]

Sigfox offre une connectivité à longue portée tout en maintenant une consommation d'énergie extrêmement faible. Cette technologie est idéale pour les dispositifs IdO qui nécessitent la transmission de petites quantités de données sur de grandes distances, souvent dans des environnements où la connectivité cellulaire traditionnelle est impraticable ou trop coûteuse.

#### 3.3.2.1. Caractéristique de la technologie Sigfox

Sigfox offre une connectivité à longue portée tout en maintenant une consommation d'énergie extrêmement faible. Cette technologie est idéale pour les dispositifs IdO qui nécessitent la transmission de petites quantités de données sur de grandes distances, souvent dans des environnements où la connectivité cellulaire traditionnelle est impraticable ou trop coûteuse.

#### 3.3.2.2. Les fréquences utilisées

Sigfox utilise les bandes de fréquences ISM (Industriel, Scientifique, and Medical), similaires à celles employées par d'autres technologies LPWAN (Low Power Wide Area Network), comme LoRa.

En Europe, cela inclut notamment la bande des 868 MHz [30].



Figure 15 Sigfox [7]

### 3.3.2.3. Architecture et le fonctionnement de Sigfox

#### 1-Dispositifs de IOT

- **Capteurs et actionneurs:** Équipés de modules Sigfox, ces dispositifs collectent et transmettent des données de manière intermittente.
- **Transmission de petites quantités de données :** Typiquement, les messages Sigfox sont très courts, avec une taille maximale de 12 octets pour les messages uplink (montant) et de 8 octets pour les messages downlink (descendant).

#### 2-stations de basse (Antennes)

- **Réseau d'Antennes :** les antennes sigfox reçoivent les messages des dispositifs IdO et transmettent vers le réseau.
- **Couverture étendue :** grâce à leur longue portée de la zone géographique.

#### 3-Backend sigfox

- **Serveurs de réseau :** Reçoivent les données des antennes et gèrent la distribution aux serveurs d'application des clients.
- **Traitement et stockage :** Les données sont traitées et stockées, prêtes à être utilisées par diverses applications IdO.

#### 4-Serveurs d'Application

- **Application client :** Utilisent les données collectées pour diverses applications, telles que le suivi de flotte, la gestion d'énergie, l'agriculture intelligente, etc.
  - **API :** Sigfox offre des API pour faciliter l'intégration des données dans les systèmes d'information des clients.
- [30]

Tableau 1 Les avantages des LoraWAN et Sigfox.

Avantages de LoraWAN	Avantages de SigFox
Basse consommation d'énergie	Faible consommation d'énergie
Longue Portée	Longue portée
Capacité de Support de masse	Simplicité d'utilisation
Coût réduit - Sécurité	Coût réduit

### 3.3.3. L'idée de full Duplex et les inconvénients de HTTP

#### 3.3.3.1. Le Full Duplex

Le full duplex fait référence à une méthode de communication où les deux parties peuvent envoyer et recevoir des données simultanément, permettant une interaction bidirectionnelle en temps réel entre les appareils ou les applications connectés. Cette méthode est souvent mise en œuvre à l'aide de protocoles tels que les Web Sockets, qui permettent aux clients et aux serveurs de communiquer via des canaux full duplex sur des connexions de longue durée. Ce type de communication est avantageux pour la fonctionnalité en temps réel et entraîne généralement moins de surcharge par rapport au polling HTTP traditionnel.

Ainsi, désigne une méthode de communication où les deux parties peuvent envoyer et recevoir des données simultanément. Cela permet une interaction bidirectionnelle en temps réel entre les dispositifs ou applications connectés.

#### 3.3.3.2. Inconvénient de http

Ce protocole de communication de base utilisé pour transférer des données sur le Web. Bien qu'il soit largement utilisé, il présente certains inconvénients par rapport aux exigences modernes de sécurité et de performance. [3]

Tableau 2 Les avantages de full Duplex et Les inconvénients de http

Avantages de Full Duplex	Inconvénients de HTTP
Communication simultanée	Communication Synchrone non adaptée à l'IdO
Optimisation de la bande passante	Lourd
Applications en Temps Réel	Presque pas de temps réel
Fluidité de la communication	Engorgement du réseau
	Performance limitée
	Incompatibilité avec les nouvelles technologies

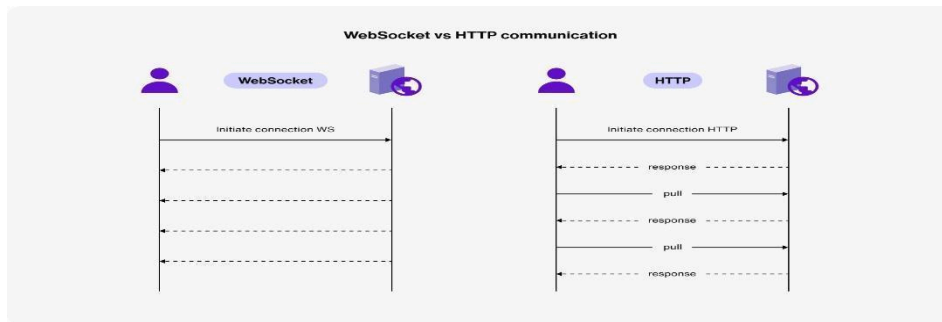


Figure 16 L'idée de full Duplex et les inconvénients de HTTP [9]

### 3.3.4. Le modelé des services des messageries

Dans l'Internet des Objets (IdO), les services de messagerie jouent un rôle crucial en permettant la communication entre les dispositifs connectés, les passerelles, les serveurs et les applications. Ces services de messagerie facilitent l'échange de données, la gestion des événements et le contrôle des dispositifs. Le modèle des services de messagerie dans l'IdO est conçu pour être fiable, évolutif et sécurisé, répondant aux besoins spécifiques des applications IdO.

#### 3.3.4.1. Composant : du modèle des Services de messageries dans l'IdO

##### 1-Dispositifs IdO (Capteurs et Actionneurs)

**Capteurs.** Collectent des données environnementales (température, humidité, etc.) et les envoient via le service de messagerie.

**Actionneurs:** Reçoivent des commandes via le service de messagerie pour effectuer des actions (allumer une lumière, ouvrir une vanne, etc.).

##### 2-passerelles IOT

**Agrégation des Données :** Recueillent les données des capteurs et les transmettent aux serveurs ou aux plateformes IdO.

**Envoi de commandes :** Relais des commandes des serveurs vers les dispositifs actionneurs.

##### 3-Serveurs de Messagerie

**Broker :** *IdO* la réception et la distribution des messages. Un exemple typique est un broker MQTT (Message Queuing Telemetry Transport).

**Gestion des Topics:** Organise les messages en différents "topics" ou sujets, facilitant la publication/abonnement (pub/sub).

##### 4-Application

**Consommateurs de Données :** Reçoivent les données des capteurs pour les analyser et les visualiser.

**Envoi de Commandes :** Envoyent des commandes aux dispositifs IdO via le serveur de messagerie.

#### 3.3.4.2. Les protocoles de Messagerie Utilisée dans l'IdO

##### 1- MQTT (Message Queuing Telemetry Transport)

**Caractéristiques :** Léger, conçu pour des environnements à bande passante limitée, et supportant le modèle (pub/sub).

**Utilisation :** Idéal pour les applications nécessitant une faible latence et une communication fiable, comme les systèmes de domotique et les capteurs industriels.

## 2- **COAP (Constrained Application Protocol)**

**Caractéristiques :** : Léger et conçu pour les dispositifs contraints en termes de ressources, utilisant le modèle client/serveur.

**Utilisation :** Convient aux applications où les dispositifs ont des ressources limitées, comme les capteurs environnementaux.

## 3- **AMQP (Advanced Message Queuing Protocol)**

**Caractéristiques :** : Plus lourd que MQTT et CoAP, offrant des fonctionnalités avancées de file d'attente et de routage des messages

**Utilisation :** Utilisé dans des environnements nécessitant une fiabilité et une sécurité élevées, comme les transactions financières et les systèmes critiques.

## 4- **HTTP/HTTPS**

**Caractéristiques :** : Utilisé pour la communication RESTful, bien adapté aux applications web.

**Utilisation :** Utilisé pour les applications IdO où la communication n'est pas critique en temps réel et où l'interopérabilité avec les services web est nécessaire.

### 3.3.4.3. Modèle de Publication/Abonnement (PUB/SUB)

Le modèle pub/sub est couramment utilisé dans les services de messagerie IdO en raison de sa simplicité et de son efficacité. Voici comment il fonctionne.

#### 1-publication (Pub) :

- Les dispositifs IdO publient des messages sur des "topics" spécifiques au broker de messagerie.

#### 2-Abonnement (Sub) :

- Les applications ou autres dispositifs s'abonnent à ces "topics" pour recevoir les messages correspondants.

#### 3-Distribution des Messages :

- Le broker distribue les messages publiés aux abonnés correspondants, assurant une communication asynchrone et décentralisée.

	MQTT	CoAP	AMQP	HTTP
<b>Base protocol</b>	TCP	UDP	TCP	TCP
<b>Paradigm</b>	Publish/Subscribe	Request/Response or Publish/Subscribe	Publish/Subscribe or Request/Response	Request/Response
<b>Header Size</b>	2 Bytes	4 Bytes	8 Bytes	Undefined
<b>Message Size</b>	Small and Undefined (up to 256 MB)	Small and Undefined	Negotiable and Undefined	Large and Undefined
<b>Reliability</b>	QoS 0 - At most once QoS 1 - At least once QoS 2 - Exactly once	CON Message NON Message	Settle Format Unsettle Format	Limited (via TCP)
<b>Standards</b>	OASIS, Eclipse Foundations	IETF, Eclipse Foundation	OASIS, ISO/IEC	IETF and W3C
<b>Licensing</b>	Open Source	Open Source	Open Source	Free

Figure 17 Les protocoles de messagerie utilisés dans l'IdO [8]

### 3.3.4.4. Sécurité dans les Services de Messageries *IdO*

**1-Authentification et Autorisation :** Assurer que seuls les dispositifs et les utilisateurs autorisés peuvent publier ou s'abonner aux topics.

**2-Chiffrement des Données :** Utilisation de TLS/SSL pour sécuriser les communications et protéger les données en transit contre les interceptions.

**3-Intégrité des Messages :** Garantir que les messages ne sont pas modifiés en cours de route, en utilisant des signatures numériques ou des hachages cryptographiques.

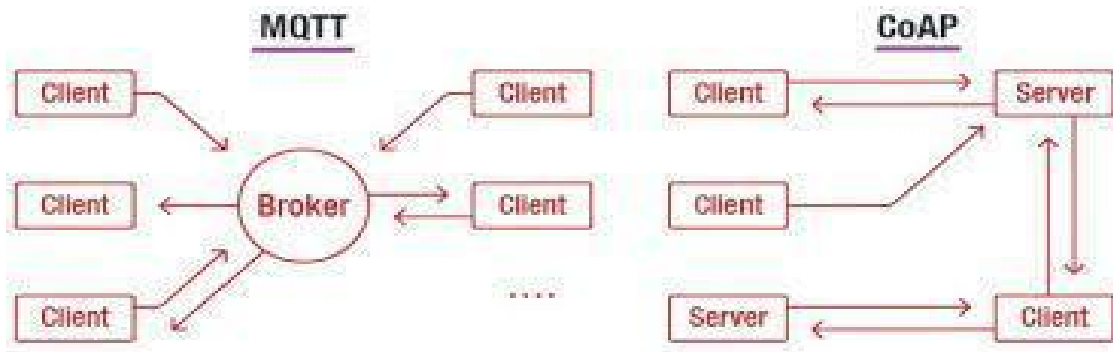


Figure 18 Les protocoles de Messagerie MQTT et CoAP l'IOT [10]

### 3.3.5. MQTT

#### 3.3.5.1. Transport de télémétrie de file d'attente de messages (MQTT)

MQTT a démarré en tant que protocole propriétaire IBM développé par Andy Stanford-Clark et Arlen Nipper d'Arcom Control Systems Ltd (Eurotech) en 1999. Utilisé pour communiquer avec SCADA systèmes dans l'industrie pétrolière et gazière. Il s'agit désormais d'un protocole open source supervisé par l'Organisation pour l'avancement des normes d'information structurée (OASIS). Le MQ dans MQTT signifie « Message Queuing », cependant, il n'y a pas de file d'attente de messages dans la communication MQTT. Le protocole fournit désormais une messagerie de publication et d'abonnement et est devenu populaire auprès des utilisateurs.

Aujourd'hui, MQTT est l'un des principaux protocoles open source utilisés pour connecter les dispositifs de l'Internet des Objets . En plus de MQTT, il existe d'autres protocoles de messagerie populaires qui prennent en charge les applications IdO. Ceux-ci incluent le message avancé Protocole de file d'attente (AMQP), protocole d'application contraint (CoAP), messagerie extensible et Protocole de présence (XMPP), service de distribution de données (DDS), ZigBee et Zwave. [34].

#### 3.3.5.2. Le principe de MQTT

Le protocole MQTT fonctionne sur les principes du modèle de publication /abonnement. Dans la communication réseau traditionnelle, les clients et les serveurs communiquent directement entre eux. Les clients demandent des ressources ou des données au serveur, puis le serveur les traite et renvoie une réponse. Cependant, MQTT utilise un modèle de publication/abonnement pour découpler l'expéditeur du message (éditeur) du récepteur du message (abonné).



Ces services de messagerie facilitent l'échange de données, la gestion des événements et le contrôle des dispositifs. Le modèle des services de messagerie dans l'IdO est conçu pour être fiable, évolutif et sécurisé, répondant aux besoins spécifiques des applications IdO [4].

### 3.3.5.3. Fonctionnement de MQTT

Un aperçu du fonctionnement de MQTT est donné ci-dessous.

- Un client MQTT établit une connexion avec l'agent MQTT.
- Une fois connecté, le client peut soit publier des messages, soit s'abonner à des messages spécifiques, soit faire les deux.
- Lorsque l'agent MQTT reçoit un message, il le transmet aux abonnés qui sont intéressés

### 3.3.5.4. Exemples d'Utilisation de MQTT

#### 1. Domotique

- Contrôle et surveillance des appareils domestiques connectés, comme les lumières, les thermostats, et les systèmes de sécurité.

#### 2. Surveillance Environnementale

- Collecte et transmission de données de capteurs environnementaux, tels que la température, l'humidité, et la qualité de l'air.

#### 3. Industrie 4.0

- Suivi et gestion des machines et équipements industriels pour la maintenance prédictive et l'optimisation des processus

#### 4. Santé Connectée :

- Surveillance des signes vitaux des patients et transmission des données vers les plateformes de santé pour analyse

### 3.3.5.5. Rubrique MQTT

Le terme « rubrique » (topic) fait référence aux mots-clés que l'agent MQTT utilise pour filtrer les messages destinés aux clients MQTT. Les rubriques sont organisées de manière hiérarchique, à l'instar d'un répertoire de fichiers ou de dossiers. Par exemple, considérons un système de maison intelligente fonctionnant dans une maison à plusieurs niveaux qui possède différents appareils intelligents à chaque étage. Dans ce cas, l'agent MQTT peut organiser les rubriques comme suit :

*maison /rez-de-chaussée/salon/lumière*

*maison /premier étage /cuisine/température*

### 3.3.5.6. Publication MQTT

Le client détermine le format des données, comme les données textuelles, les données binaires, les fichiers XML ou JSON. Par exemple, une lampe dans le système de maison intelligente peut publier un message *allumer* pour la rubrique *salon/lumière*. [34]

### 3.3.5.7. Abonnement MQTT

Les clients MQTT envoient un message *SUBSCRIBE* (S'ABONNER) à l'agent MQTT, pour recevoir des messages sur les rubriques qui les intéressent. Ce message contient un identifiant unique et une liste

d'abonnements. Par exemple, l'application de maison intelligente sur votre téléphone veut afficher le nombre d'ampoules allumées dans votre maison. Il s'abonnera à la rubrique *lumière* et augmentera le compteur pour tous les messages *allumer*. [34]

### 3.3.5.8. Sécurité dans MQTT

Pour assurer une communication sécurisée entre les clients et l'agent MQTT, plusieurs mécanismes peuvent être mis en place :

#### 1. Authentification et Autorisation :

- Utilisation de noms d'utilisateur et de mots de passe pour authentifier les clients.
- Implémentation de contrôles d'accès pour restreindre la publication et l'abonnement à certains topics.

#### 2. Chiffrement : Utilisation de TLS/SSL pour chiffrer les communications entre les clients et le broker, protégeant les données en transit contre les interceptions.

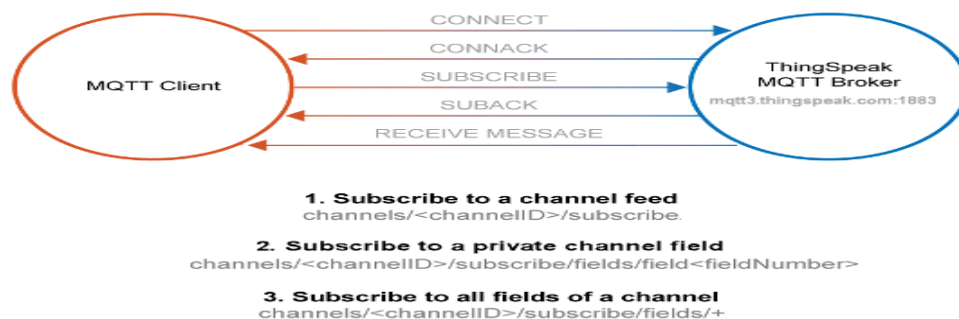


Figure 19 Transport de télémétrie de file d'attente de messages [11]

### 3.3.6. Le protocole MQTT

Nous avons déjà défini Mqtt dans la section sur le protocole de messagerie où nous avons parlé de son histoire. De plus, cette section met l'accent et fourni plus de détails :

Premières choses à savoir, MQTT convient pour être utilisé sur des appareils ayant des capacités limitées en termes de bande passante et de transmission de données. Il est basé sur un modèle de publication/abonnement qui doit être pris en compte en tant que client MQTT, alors que Broker est un serveur MQTT, adapté au M2M (machine to machine) et WSN (Réseaux de capteurs sans fil)

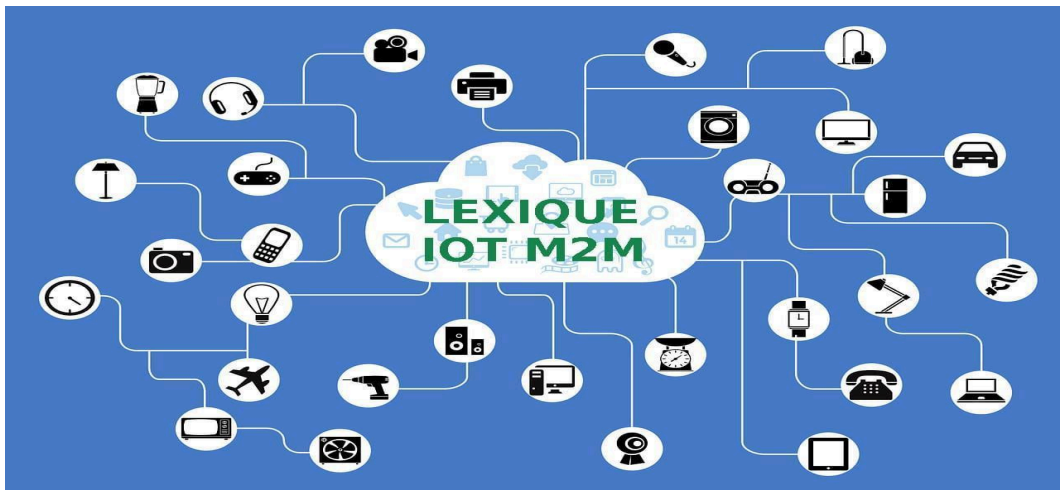


Figure 20 IOT M2M [12]

### 3.3.6.1. Concepts de base de MQTT

En termes simples, les composants de base de Mqtt sont les suivants :

**1-client MQTT :** Un client MQTT est tout dispositif capable d'exécuter une bibliothèque MQTT. Les clients peuvent être des serveurs puissants ou des microcontrôleurs à faible puissance. Selon les actions qu'ils effectuent, les clients MQTT peuvent être classés comme

**Editeur (Publisher) :** Un client qui envoie des messages à un ou plusieurs topics.

**Abonné (Subscriber) :** Un client qui reçoit des messages publiés sur des topics auxquels il est abonné.

**2-Agent MQTT(Broker) :** L'agent MQTT, ou broker (courtier), est le système dorsal qui coordonne les messages entre les différents clients MQTT. Le broker a plusieurs responsabilités :

1. Réception et filtrage des messages : Le broker reçoit les messages publiés par les clients et les filtre en fonction des topics.
2. Gestion des Abonnements : Identifie les clients abonnés à chaque topic et transmet les messages appropriés à ces clients
3. Authentification et Autorisation : Vérifie l'identité des clients et contrôle leur accès aux différents topics.
4. Gestion des Sessions : Traite les connexions, les déconnexions, et les messages manqués.
5. Transmission des Messages : Peut envoyer les messages reçus à d'autres systèmes pour une analyse plus approfondie.

#### Exemples de Brokers MQTT

On a utilisé le broker Mosquitto. Il est un broker MQTT, open-source et largement utilisé surtout dans les applications dédiées à la domotique.

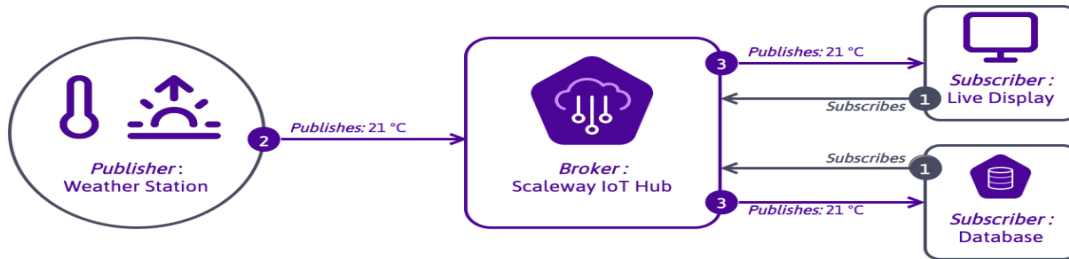


Figure 21 Concept de base de MQTT [13]

### 3.3.6.2. Sujets et abonnements

Dans MQTT, un éditeur envoie (publie) un message sous un nom de sujet. Par la suite, tous les abonnés sous le nom du sujet reçoivent le message via un courtier. On peut avoir autant de sujets comme nous le souhaitons, chacun ayant des clients différents à chaque extrémité. Cela peut être un à un, plusieurs à plusieurs ou n'importe quelle combinaison. L'architecture de publication/abonnement est très flexible et évolutive, ce qui permet qu'elle soit idéale pour les applications IdO.

### Niveaux de qualité de service

La QoS est une fonctionnalité clé du protocole MQTT qui offre des qualités de service de messagerie supplémentaires qui garantissent que le message en transit est livré comme requis par le service. Cela donne au client le pouvoir de choisir un niveau de service qui correspond à la fiabilité de son réseau et à sa logique applicative, car MQTT gère la retransmission des messages et garantit la livraison d'un message.

Bien que TCP/IP offre une présentation garantie des données, une perte de données peut survenir si une connexion TCP est cassé et les messages transmis sont perdus. Par conséquent, MQTT ajoute trois niveaux de QoS en haut de TCP pour transmettre des messages [4].

Tableau 3 Différents types de qualité de services [4]

QoS 0	QoS 1	QoS 2
<ul style="list-style-type: none"> <li>-Uniquement la garantie TCP</li> <li>-Souvent appelé « feu et oublier », la livraison d'un message n'est pas garantie.</li> <li>-Le destinataire n'accuse pas réception de message.</li> <li>-N'est pas stocké et retransmis par l'expéditeur.</li> </ul>	<ul style="list-style-type: none"> <li>-Garantie MQTT avec confirmation</li> <li>-Le plus fréquemment utilisé parce qu'il garantit qu'un message est livré au moins une fois au récepteur, mais permet plusieurs livraisons.</li> <li>-L'expéditeur stocke le message jusqu'à ce qu'il reçoive un accusé de réception.</li> </ul>	<ul style="list-style-type: none"> <li>-Garantie MQTT</li> <li>Avec poignée de main</li> <li>-Le plus haut niveau de services en MQTT</li> <li>-Le message est envoyé précisément une fois,</li> <li>-Il garantit que chaque message est reçu une seule fois par le récepteur prévu.</li> <li>-Le plus sûr mais le plus lent</li> </ul>

### 3.3.6.3. Messages conservés

Chaque client qui souscrit à un modèle de sujet qui correspond au sujet du message conservé reçoit immédiatement le message conservé après leur inscription. Le courtier ne stocke qu'un seul message conservé par sujet.

### 3.3.6.4. Testaments

Une autre fonctionnalité supplémentaire du serveur MQTT est le Dernier Testament (LWT). C'est un message utilisé pour notifier la raison pour laquelle une connexion entre un courtier et un client, qui souscrivait activement ou la publication sur un sujet a été interrompue. Le message de notification est envoyé à tous les clients abonnés à ce sujet, afin que tout client autorisé dans le système puisse publier une nouvelle valeur sur le réseau périphérique, LWT n'est pas seulement généré par l'absence de réponse de maintien en vie, mais également au cas où un client se terminerait une connexion sans déconnexion appropriée du courtier.

### 3.3.6.5. Keep Alive et prise de contrôle des clients

**a) Keep Alive** : garantit que la connexion entre le courtier et le client est toujours ouverte et que le courtier et le client sont conscients d'être connectés. Lorsque le client établit une connexion au courtier, le client communique un intervalle de temps en secondes au courtier.

Cet intervalle définit la durée maximale pendant laquelle le courtier et le client ne peuvent pas communiquer entre eux.

**b) Prise de contrôle du client** : généralement, un client déconnecté tente de se reconnecter. Parfois, le courtier a toujours une connexion semi-ouverte avec le client. Dans MQTT, si le courtier détecte une entrée semi-ouverte, il effectue une « prise de contrôle client ». Le courtier ferme la connexion précédente au même client (déterminée par l'identifiant client), et établit une nouvelle connexion avec le client. Ce comportement garantit que la connexion semi-ouverte n'arrête pas la connexion déconnectée.

### 3.3.6.6. Messages de session persistante et de file d'attente

Pour recevoir des messages d'un courtier MQTT, un client se connecte au courtier et crée des abonnements aux sujets qui l'intéressent. Si la connexion entre le client et le courtier est interrompue lors d'une session non persistante, ces sujets sont perdus et le client doit se reconnecter. Se réabonner à chaque fois que la connexion est interrompue est un fardeau pour les clients contraints et disposant de ressources limitées. Pour éviter ce problème, le client peut demander une session persistante lorsqu'il se connecte au courtier. Les sessions persistantes enregistrent toutes les informations pertinentes pour le client sur le courtier via l'identifiant que le client fournit lorsqu'il établit une connexion avec le courtier [4].

## 3.3.7. Mosquitto Broker (Serveur)

Eclipse Mosquitto est un courtier de messages open source qui implémente le protocole MQTT versions 3.1 et 3.1.1. C'est l'un des courtiers MQTT les plus connus.

Il est très simple à installer et à utiliser. Il est léger et peut être utilisé sur tous les appareils, des ordinateurs mono carte basse consommation aux serveurs complets. Le protocole MQTT fournit une méthode légère d'exécution de messagerie à l'aide d'un modèle de publication/abonnement. Cela fait adapter à la messagerie Internet des objets, par exemple avec des capteurs basse consommation ou des appareils mobiles tels que les téléphones, les ordinateurs embarqués ou les microcontrôleurs. Le projet Mosquitto fournit également une bibliothèque C pour implémenter les clients MQTT, et ceux les plus populaires sont `mosquitto_pub` et `mosquitto_sub`

Clients MQTT en ligne de commande. Il peut être installé sur des machines UNIX. Il peut être sécurisé via SSL et mots de passe.

### *1- Son importance*

- Après avoir configuré le serveur mosquitto MQTT sur un Raspberry Pi. Ensuite, cela vous permet de connecter tous Nœuds DIY vers le courtier MQTT.
- Dans MQTT, le courtier est principalement responsable de la réception de tous les messages, du filtrage des messages, puis publie le message à tous les clients abonnés.
- Pour communiquer avec l'ESP8266 et esp32 via Wifi (ou depuis Internet), le protocole de communication MQTT (Mosquitto Broker) est utilisé. Il reçoit les messages des éditeurs et veille à leur bonne maintenance. Afin que seuls les éditeurs disposant d'une authentification appropriée puissent envoyer des messages. Le Courtier fournit ensuite les données dans « Sujets » aux « Abonnés ». Dans le même esprit, le Courtier vérifie l'authenticité de chaque « Abonné » et ne fournit à chaque Abonné que les sujets il est abonné.
- Mosquitto appartient à la catégorie des logiciels Messaging Oriented Middleware (MOM) qui prend en charge l'envoi et la réception de messages entre les systèmes distribués.

## **2- Server ports: Le serveur écoute sur les ports suivants**

- 1883: MQTT, unencrypted, unauthenticated (SSL).
- 1884: MQTT, unencrypted, authenticated.
- 8883 : MQTT, encrypted, unauthenticated (SSL).
- 8884: MQTT, encrypted, client certificate required (SSL).
- 8885: MQTT, encrypted, authenticated.
- 8887 : MQTT, encrypted, server certificate deliberately expired.
- 8080: MQTT over WebSockets, unencrypted, unauthenticated.
- 8081 : MQTT over WebSockets, encrypted, unauthenticated.
- 8090: MQTT over WebSockets, unencrypted, authenticated.
- 8091: MQTT over WebSockets, encrypted, authenticated.

Mais les ports standard les plus couramment utilisés par Mqtt sont le port TCP/IP 1883. Le port TCP/IP 8883 est également enregistré, par exemple en utilisant MQTT sur SSL.

Les auditeurs authentifiés nécessitent un nom d'utilisateur/mot de passe :

- **rw / readwrite** : read/write access to the # topic hierarchy
- **ro / readonly** : read only access to the # topic hierarchy
- **wo / writeonly** : write only access to the # topic hierarchy

[33]

## **Conclusion**

Le choix de MQTT comme principal protocole de communication entre appareils a été jugé approprié pour un système de maison intelligente à base d'IdO, revient à son utilisation idéale dans des environnements contraints. De plus, il n'est pas exigeant en termes de consommation, avec une capacité de calcul et une mémoire limitées, une bande passante limitée et une simplicité d'utilisation, un déploiement rapide et sa fiabilité dans le transfert de données et de commandes vers des appareils distants.

Nous avons donc consacré beaucoup de temps à comprendre son fonctionnement, son architecture, les domaines dans lesquels il est principalement utilisé et ses différents courtiers avec leurs limitations et fonctionnalités.

# CHAPITRE 04 Implémentation

## 4.1. INTRODUCTION

Une maison intelligente signifie que la maison contient un système qui communique avec des appareils pour effectuer des tâches spécifiques, et n'importe qui peut utiliser le système de maison intelligente pour programmer plusieurs de ses outils, tels que la configuration et la surveillance du système de sécurité domestique et des caméras, ou alerter de la présence d'un incendie ou contrôler des appareils tels que le réfrigérateur, le climatiseur, etc. Les appareils d'une maison intelligente sont connectés les uns aux autres via Internet, comme le montre la figure 22, permettant à l'utilisateur de contrôler à distance des fonctions telles que l'accès sécurisé à la maison, température et éclairage. Il présente également de nombreux avantages, notamment ce qui suit :

- Augmentez le confort et gagnez du temps.
- Contrôle complet avec un seul appareil.
- Envoi de notifications en cas de problème ou de danger.



Figure 22 Système de sécurité domestique. [40]

### Objectif

Pour concevoir ce système proposé, nous devons nous appuyer principalement sur de nombreux outils que nous présenterons dans ce qui suit :

## 4.2 Les langages utilisés

### 4.2.1. Langage HTML (HyperText Markup Language)

#### Introduction

HTML (HyperText Markup Language) est le langage standard utilisé pour créer et structurer des pages web. Il permet de définir le contenu et la structure des documents web à l'aide d'éléments et d'attributs. HTML est le fondement de tout site web et est utilisé en conjonction avec CSS (Cascading Style Sheets) pour la mise en forme et JavaScript pour les interactions dynamiques. Un document HTML typique commence par une déclaration de type de document et est structuré en deux parties principales : l'en-tête (head) et le corps (body).[23]

#### Éléments HTML de Base

##### 1. Éléments de Texte :

<h1> à <h6> : Titres, du plus grand au plus petit./ <p> : Paragraphe./ <a> : Lien hypertexte (attribut Href pour l'URL)./ <span> : Conteneur en ligne sans signification particulière, utilisé pour la mise en forme. / <div> : Conteneur de bloc sans signification particulière, utilisé pour structurer le contenu.

##### 2. Éléments de Liste :

<ul> : Liste non ordonnée (à puces). / <ol> : Liste ordonnée (numérotée)./ <li> : Élément de liste.

##### 3. Éléments de Table :

<table> : Table. /<tr> : Ligne de tableau / <td> : Cellule de tableau/ <th> : Cellule d'en-tête de tableau.

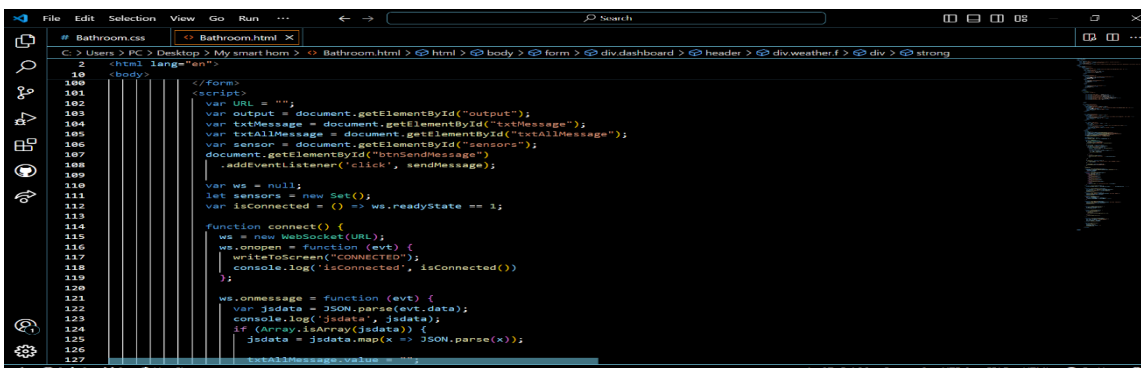
##### 4. Éléments de Formulaire :

<form> : Formulaire./ <input> : Champ de saisie./ <textarea> : Zone de texte multiligne./ <button> : Bouton. / <select> : Liste déroulante. / <option> : Option dans

une liste déroulante.

**Attributs HTML :** Les éléments HTML peuvent avoir des attributs qui fournissent des informations supplémentaires. Les attributs sont placés dans la balise d'ouverture de l'élément.

Id : Identifiant unique pour l'élément. / class : Classe(s) CSS appliquée(s) à l'élément. /src : URL d'une source externe, souvent utilisé pour les images et les scripts. Href : URL pour les liens hypertextes. Alt : Texte alternatif pour les images.



```
100 </form>
101 </body>
102 </html lang="en">
103
104 <script>
105 var URL = "";
106 var output = document.getElementById("output");
107 var txtMessage = document.getElementById("txtMessage");
108 var txtAllMessage = document.getElementById("txtAllMessage");
109 var sensor = document.getElementById("sensors");
110 document.getElementById("btnSendMessage");
111 addEventListener("click", sendMessage);
112
113 var ws = null;
114 let sensors = new Set();
115 var isConnected = () => ws.readyState == 1;
116
117 function connect(URL) {
118   ws = new WebSocket(URL);
119   ws.onopen = function (evt) {
120     writeToScreen("CONNECTED");
121     console.log("isConnected", isConnected());
122   };
123
124   ws.onmessage = function (evt) {
125     var jsonData = JSON.parse(evt.data);
126     console.log("jsonData", jsonData);
127     if (!Array.isArray(jsonData)) {
128       jsonData = jsonData.map(x => JSON.parse(x));
129     }
130   };
131 }
132
133 function sendMessage() {
134   if (ws) {
135     ws.send(txtMessage.value);
136   }
137 }
138
139 function writeToScreen(message) {
140   txtAllMessage.value += message + "\n";
141 }
142
143 function init() {
144   connect(URL);
145 }
146
147 init();
148
149 </script>
150 </body>
151 </html>
```

Figure 23 Fragment de code HTML.



## 4.2.2. Langage (SCC)

CSS, ou Cascading Style Sheets, est un langage de feuille de style utilisé pour décrire la présentation et la mise en forme des documents HTML. Grâce aux CSS, vous pouvez contrôler l'apparence des pages web, en définissant des styles pour les éléments HTML comme les polices, les couleurs, les espacements, les dispositions, et bien plus encore. Les CSS permettent de séparer le contenu (HTML) de la présentation, facilitant ainsi la maintenance et l'évolution des sites web. [31]

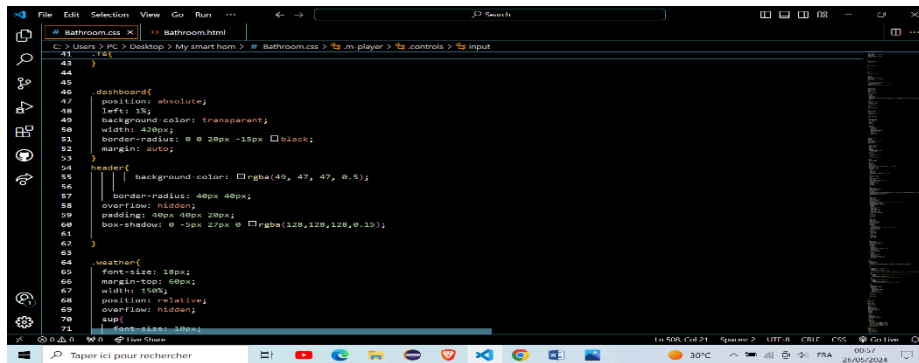


Figure 24 Fragment de code CSS.

## 4.2.3. Langage (JS)

JavaScript est un langage de script, multiplateforme et orienté objet. C'est un langage léger qui doit faire partie d'un environnement hôte (un navigateur web par exemple) pour qu'il puisse être utilisé sur les objets de cet environnement, JavaScript contient une bibliothèque standard d'objets tels que Array, Date, et Math, ainsi qu'un ensemble d'éléments de langage tels que les opérateurs, les structures de contrôles et les instructions. Ces fonctionnalités centrales et natives de JavaScript peuvent être étendues de plusieurs façons en fournissant d'autres objets, par exemple :

JavaScript côté client étend ces éléments de base en fournissant des objets pour contrôler le navigateur et le Document Object Model (DOM). Par exemple, les extensions du langage côté client permettent de placer des éléments dans un formulaire HTML, de réagir aux événements déclenchés par l'utilisateur (les clics, la saisie d'un formulaire, les actions de navigation, etc.). [32]

```
<!DOCTYPE html>
<html>
  <head>
    <title>Cours JavaScript</title>
    <meta charset='utf-8'>
    <link rel='stylesheet' href='cours.css'>
    <script type='module' src='cours.js'></script>
    <script type='module' src='cours2.js'></script>
  </head>
  <body>
    <h1>Modules</h1>
    <p id='p1'></p>
    <p id='p2'></p>
    <p id='p3'></p>
  </body>
</html>
```

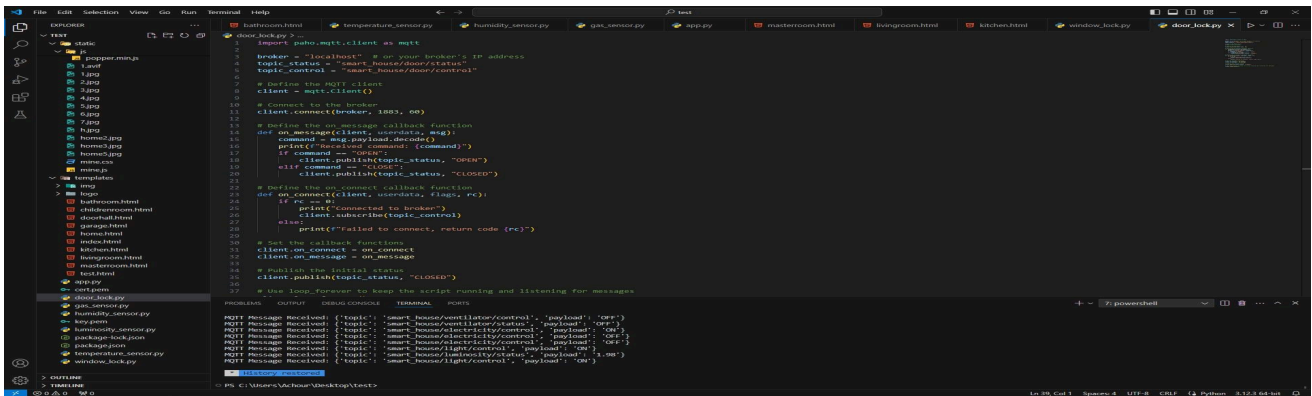
Figure 25 Fragment de code java script.

## Avantages de JavaScript

1. **Interactivité** : JavaScript permet de rendre les pages web interactives et engageantes en répondant aux actions des utilisateurs en temps réel.
2. **Flexibilité** : En tant que langage interprété, JavaScript est extrêmement flexible et peut être utilisé pour une large gamme d'applications, des simples scripts côté client aux applications serveur complexes.
3. **Compatibilité** : JavaScript est pris en charge par tous les navigateurs modernes, ce qui garantit une large compatibilité avec les différentes plateformes et appareils.
4. **Rich Ecosystem** : JavaScript bénéficie d'un écosystème riche avec une multitude de bibliothèques et Framework comme React, Angular, Vue.js pour le développement front-end, et Node.js pour le développement back-end.

## 4.2.4. Langage (python)

Créé par Guido van Rossum et publié pour la première fois en 1991. C'est un langage open-source, ce qui signifie qu'il est libre d'utilisation et que son code source est accessible à tous. Python est connu pour sa simplicité et sa facilité d'apprentissage, ce qui en fait un excellent choix pour les débutants. Il prend en charge plusieurs paradigmes de programmation, notamment la programmation procédurale, orientée objet et fonctionnelle. Python est utilisé dans une variété de domaines, y compris le développement web, la science des données, l'intelligence artificielle, l'automatisation, et bien plus encore. [19]



```
door_lock.py
import paho.mqtt.client as mqtt

broker = "localhost" # or your broker's IP address
topic_status = "smart_house/door/status"
topic_control = "smart_house/door/control"

# Define the MQTT client
client = mqtt.Client()

# Connect to the broker
client.connect(broker, 1883, 60)

# Define the on_message callback function
def on_message(client, userdata, msg):
    command = msg.payload.decode()
    print(f"Received command: {command}")
    if command == "unlock":
        client.publish(topic_status, "open")
    elif command == "close":
        client.publish(topic_status, "closed")

# Define the on_connect callback function
def on_connect(client, userdata, flags, rc):
    if rc == 0:
        print(f"Connected to broker")
        client.subscribe(topic_control)
    else:
        print(f"Failed to connect, return code {rc}")

# Set the callback functions
client.on_message = on_message
client.on_connect = on_connect
client.on_message = on_message

# Publish the initial status
client.publish(topic_status, "closed")

# Use loop_forever to keep the script running and listening for messages
mqtt.loop_forever()

# Run the script
if __name__ == "__main__":
    mqtt.loop_forever()

MQTT Message Received: {'topic': 'smart_house/ventilator/control', 'payload': 'off'}
MQTT Message Received: {'topic': 'smart_house/ventilator/status', 'payload': 'off'}
MQTT Message Received: {'topic': 'smart_house/temperature/control', 'payload': '20'}
MQTT Message Received: {'topic': 'smart_house/temperature/status', 'payload': '20'}
MQTT Message Received: {'topic': 'smart_house/airpurity/control', 'payload': 'off'}
MQTT Message Received: {'topic': 'smart_house/airpurity/status', 'payload': 'off'}
MQTT Message Received: {'topic': 'smart_house/light/control', 'payload': 'on'}
MQTT Message Received: {'topic': 'smart_house/light/status', 'payload': 'on'}
```

Figure 26 Fragment de code Python.

## Avantages de Python

**Lisibilité du Code** : La syntaxe de Python est claire et concise, ce qui facilite la lecture et la maintenance du code.

**Grande Bibliothèque Standard** : Python possède une bibliothèque standard riche qui offre de nombreuses fonctionnalités prêtes à l'emploi pour différentes tâches de programmation.

**Communauté Active** : Python a une grande communauté de développeurs qui contribuent à une multitude de bibliothèques et de frameworks, facilitant le développement de projets complexes.

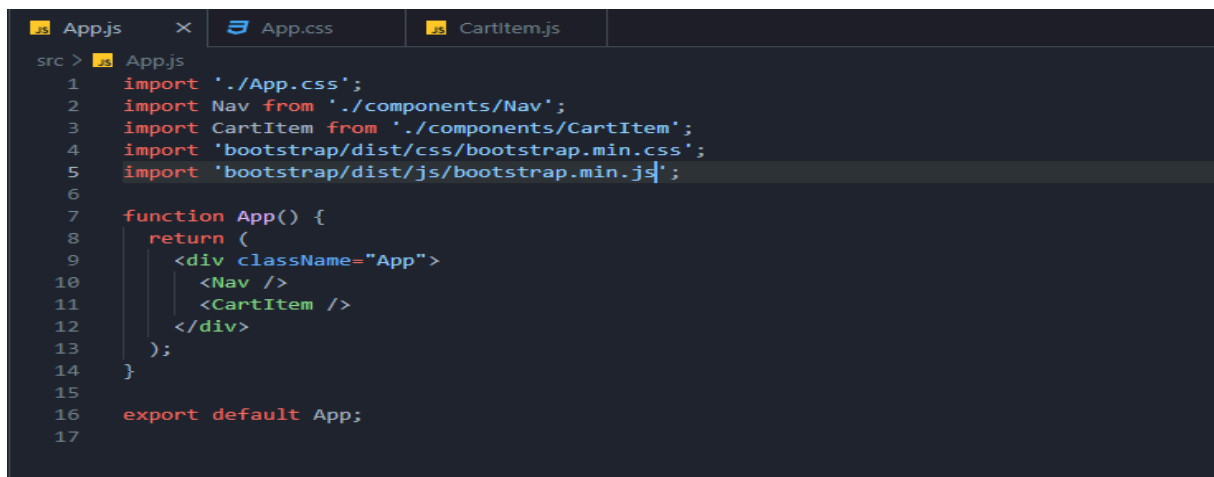
**Polyvalence** : Python est utilisé dans divers domaines, tels que le développement web (Django, Flask), la science des données (NumPy, pandas), l'intelligence artificielle (TensorFlow, PyTorch), et bien d'autres.

## 4.2.5. Bootstrap

Bootstrap est un framework CSS permettant de faciliter l'écriture de la mise en forme CSS d'un site web. Avec Bootstrap, vous gagnerez du temps et n'aurez plus besoin de vous prendre la tête sur un décalage de quelques pixels sur votre page web. Tout le CSS a déjà été pensé pour vous. Plus généralement, nous pouvons dire que c'est un rassemblement de différents morceaux de code utiles à la création d'une page web, aussi bien dans la mise en forme CSS que dans l'animation et l'interactivité du contenu JavaScript.

Bootstrap présente plusieurs avantages pour le développement tels que :

1. **Responsive** : Il garantit un site web adaptatif, indispensable à l'ère des smartphones et tablettes.
2. **Esthétique** : Propose des thèmes préfabriqués de haute qualité et de nombreuses classes CSS utilisables.
3. **Gain de temps** : Permet de se concentrer sur le code HTML en utilisant des classes CSS existantes, accélérant la mise en forme des pages web.
4. **Compatibilité** : Assure une présentation visuelle uniforme sur tous les navigateurs.
5. **Graphisme** : Simplifie le travail visuel pour les développeurs back-end, permettant de créer une apparence de site sans écrire de CSS. [39]



```
src > App.js
1 import './App.css';
2 import Nav from './components/Nav';
3 import CartItem from './components/CartItem';
4 import 'bootstrap/dist/css/bootstrap.min.css';
5 import 'bootstrap/dist/js/bootstrap.min.js';
6
7 function App() {
8   return (
9     <div className="App">
10       <Nav />
11       <CartItem />
12     </div>
13   );
14 }
15
16 export default App;
17
```

Figure 27 Fragment de code de bootstrap.

## 4.2.6. MQTT Bibliothèque

La bibliothèque core MQTT est conforme à la norme de protocole MQTT 3.1.1. Cette bibliothèque a été optimisée pour une faible empreinte mémoire. La conception de cette bibliothèque englobe différents cas d'utilisation, allant des plates-formes aux ressources limitées utilisant uniquement des messages MQTT PUBLISH QoS 0 (qualité de service 0) aux plates-formes riches en ressources utilisant QoS 2 MQTT PUBLISH sur des connexions TLS (Transport Layer Security). La bibliothèque fournit un menu de fonctions composables, dont une combinaison peut être choisie pour s'adapter précisément à un cas d'utilisation spécifique.

La bibliothèque fournit une API de haut niveau pour se connecter à un courtier MQTT, s'abonner ou se désabonner à un sujet, publier un message sur un sujet et recevoir des messages entrants. La bibliothèque expose également une API de sérialiseur/désérialiseur de bas niveau. Cette API de bas niveau gère le formatage et l'analyse des messages, laissant à l'application un contrôle total et sans surcharge sur la connexion réseau au courtier MQTT.

La bibliothèque est découplée des pilotes réseau sous-jacents via une interface de transport d'envoi et de réception à deux fonctions. Le rédacteur de l'application peut sélectionner une interface de transport existante ou implémenter la sienne, en fonction de son application.

La bibliothèque est écrite en C et conçue pour être conforme aux normes ISO C90 et MISRA C:2012. La bibliothèque n'a aucune dépendance sur des bibliothèques supplémentaires autres que la bibliothèque C standard [38].

## 4.3. Les diagrammes UML

### 4.3.1. La méthode utilisée dans la conception

Le langage de modélisation unifié (UML) peut capturer un éventail impressionnant de processus et les structures liées aux affaires et aux logiciels. UML a une telle puissance qu'un modélisateur peut l'utiliser pour l'architecture générale de toute construction présentant à la fois une structure statique et un comportement dynamique. Un projet peut s'appuyer sur UML comme langage standard pour exprimer les exigences, la conception du système, les instructions de déploiement et la structure du code.

Le praticien UML peut capturer des idées à l'aide d'outils visuels, partager efficacement ces idées avec d'autres et réagir efficacement au changement. Pour atteindre ces capacités étendues, le langage possède des fonctionnalités qui ne s'appliquent pas à toutes les circonstances. Sur un projet en utilisant UML, le modélisateur mettra l'accent sur un sous-ensemble des diagrammes potentiels. Cependant, à savoir sur quelles fonctionnalités mettre l'accent sur un projet [29].

### Modélisation du système

Pour réaliser le processus de modélisation de cette conception, nous passons par quelques étapes :

### 4.3.2 Représentants du système et tâches

Le système proposé pour la conception de maisons intelligentes se compose de quatre représentants, chacun qui a un ensemble de tâches spécifiques :

Tableau 4 Représentants du système et tâches.

Acteur	Tâche
Utilisateur	- Surveiller l'état de la maison - Contrôler l'accès à la maison en cas de risque de fuite de gaz ou incendie
Detector de gaz	Détection d'une fuite de gaz
Détecteur de température	Détecter les changements de température
Détecteur d'humidité	Détecter les changements d'humidité
Détecteur de lumière	Détecter les changements dans les capteurs de lumière
Actionneur	- Recevoir, lire et traiter les données reçues des capteurs. - Alerter sur l'existence d'un danger en envoyant un message au dispositif d'alarme.

### 4.3.3. Diagramme de cas d'utilisation

Dans le diagramme de cas d'utilisation UML, chaque cas d'utilisation est une fonction système. Généralement, un cas d'utilisation est défini comme un cours de base et plusieurs cours alternatifs en phase d'analyse des besoins.

Un cours de cas d'usage décrit une séquence d'interactions entre acteurs avec le système, qui est un modèle abstrait d'une famille de scénarios [26]. La figure 28 montre le général cas schéma du système à concevoir.

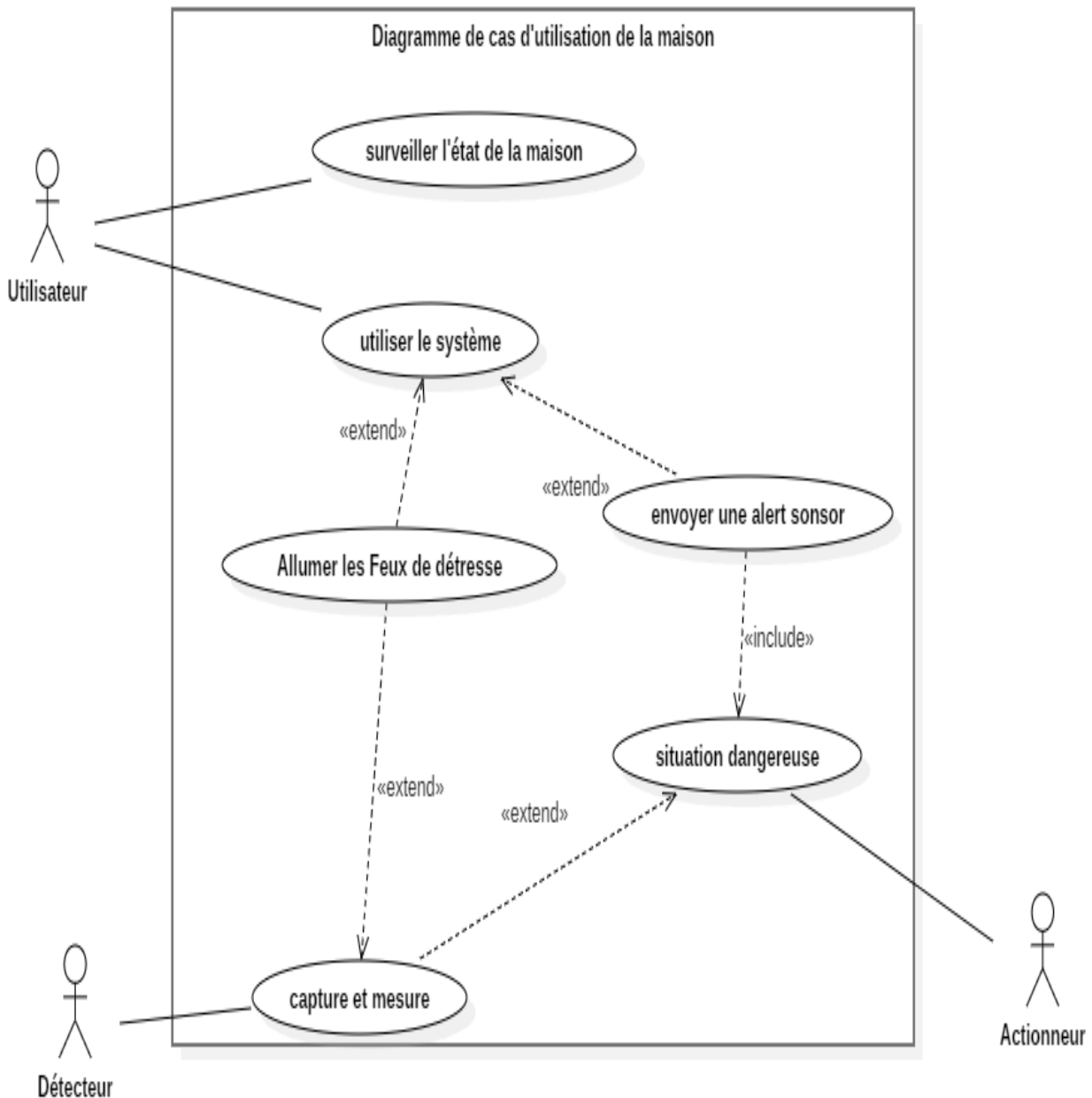


Figure 28 Diagramme de cas d'utilisation de la maison intelligente.

#### 4.3.4. Diagramme de classes

Les diagrammes de classes UML nous permettent de modéliser de manière déclarative, avec une structure cohérente avec le domaine d'application en termes de concepts et de relations entre eux [28]. Ces diagrammes sont les diagrammes les plus courants trouvés dans la modélisation de systèmes orientés objet. Dans ce travail, nous modéliserons le système à l'aide d'un diagramme de classes, comme le montre la figure 29 suivante :

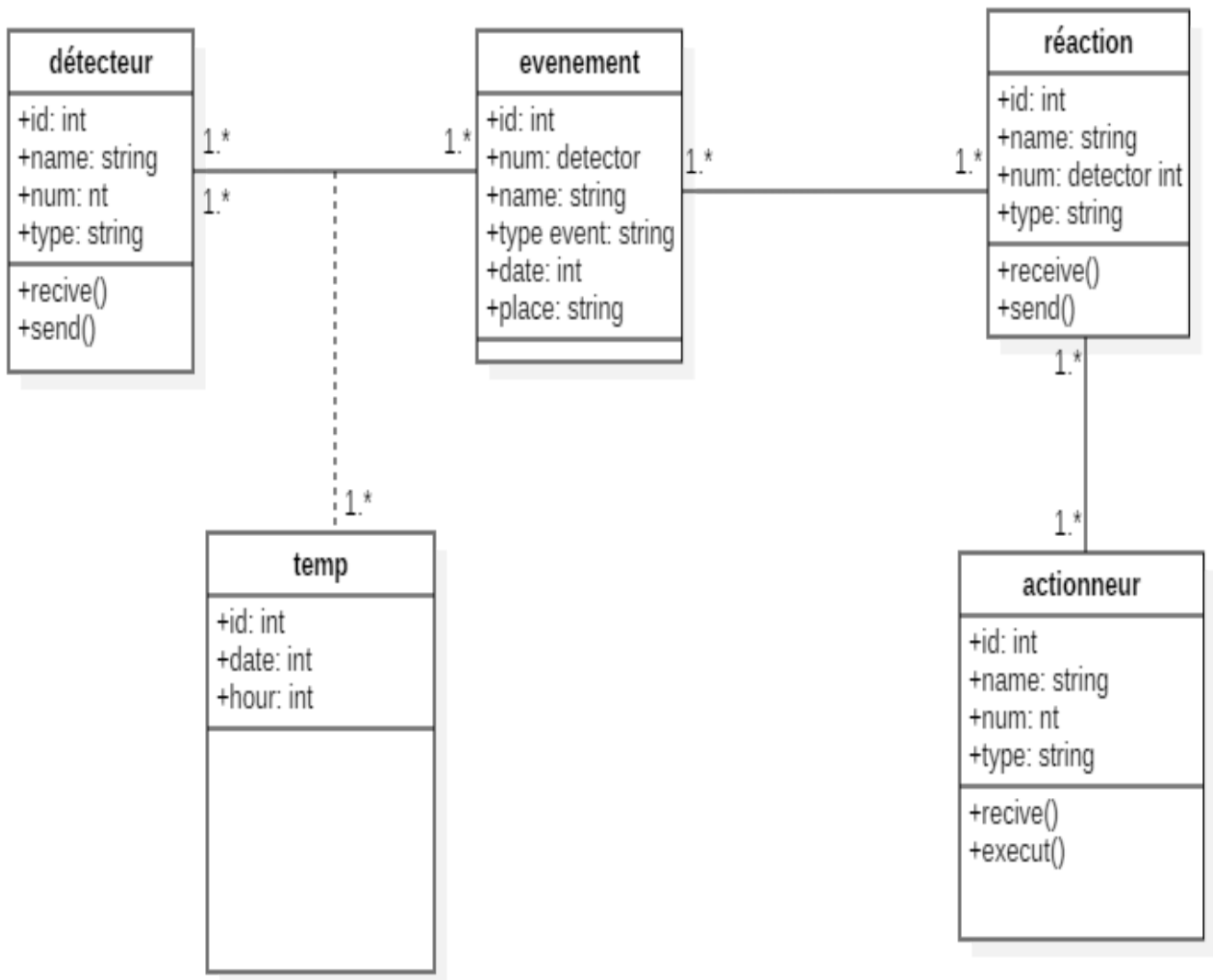


Figure 29 Classe de diagrammes

#### 4.3.5. Diagramme de séquence

Les diagrammes de séquence sont utilisés pour présenter le comportement dynamique de la conception du système. Le diagramme de séquence montre les interactions entre des objets disposés dans une séquence temporelle. Le diagramme de séquence est présenté lors de la phase de conception du système pour réaliser l'utilisation correspondante cours de cas [26].

#### 4.3.6. Diagramme d'activité

Les diagrammes d'activités sont utilisés pour modéliser et décrire un cas d'utilisation. Où les acteurs impliqués sont identifiés et le flux des événements est décrit sur la base du diagramme de cas d'utilisation, les éléments requis pour répondre aux exigences fonctionnelles respectives et les réponses du système lors de son exécution sont identifiés. Il est possible d'identifier les objets impliqués dans une activité et déterminer comment leur rôle et leur état vont changer [27].

## Représentation des diagrammes de cas, d'activités et de séquence

### - L'utilisateur

Dans le cas de la surveillance du système, l'utilisateur peut :

- Explorez le système domestique
- Recevoir des notifications du système
- Identifier le type de danger.

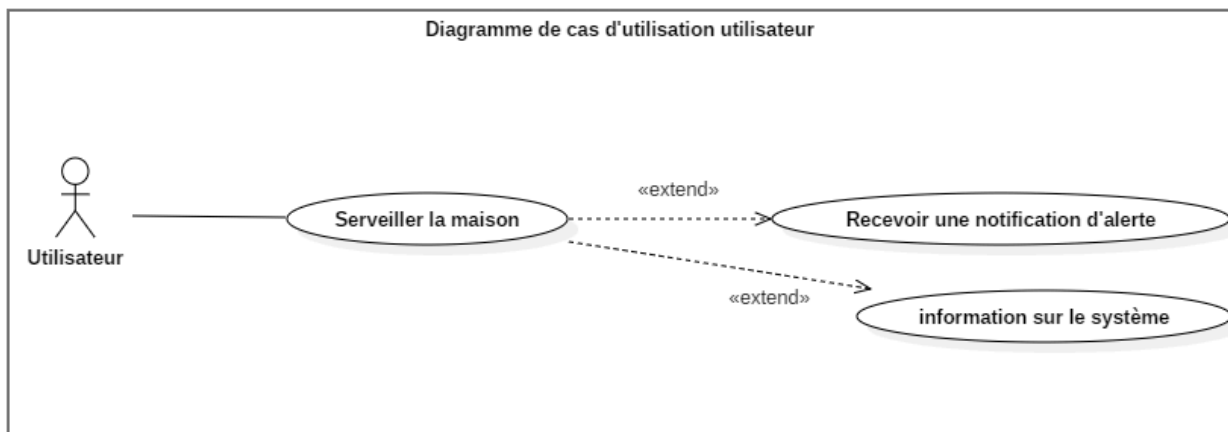


Figure 30 Diagramme de cas d'utilisation utilisateur.



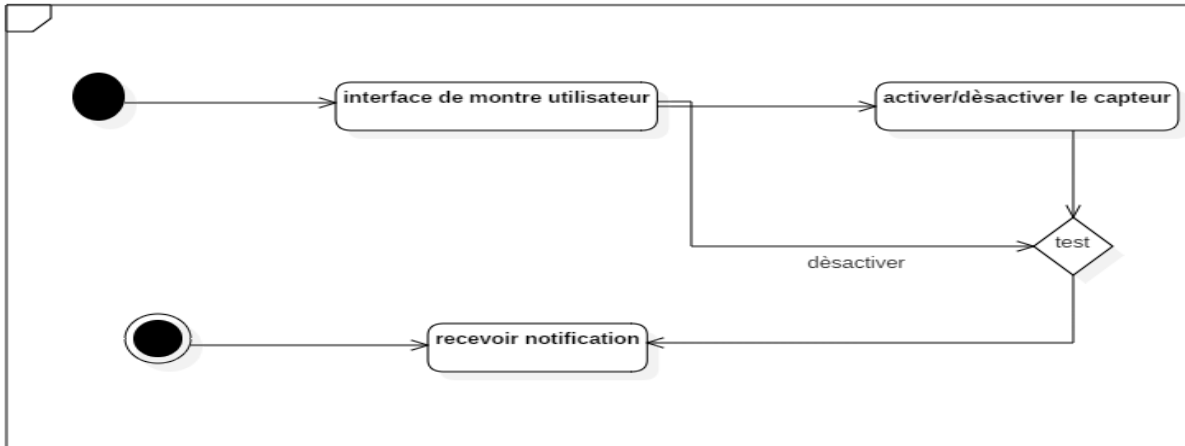


Figure 31 Diagramme d'activité pour l'utilisateur.

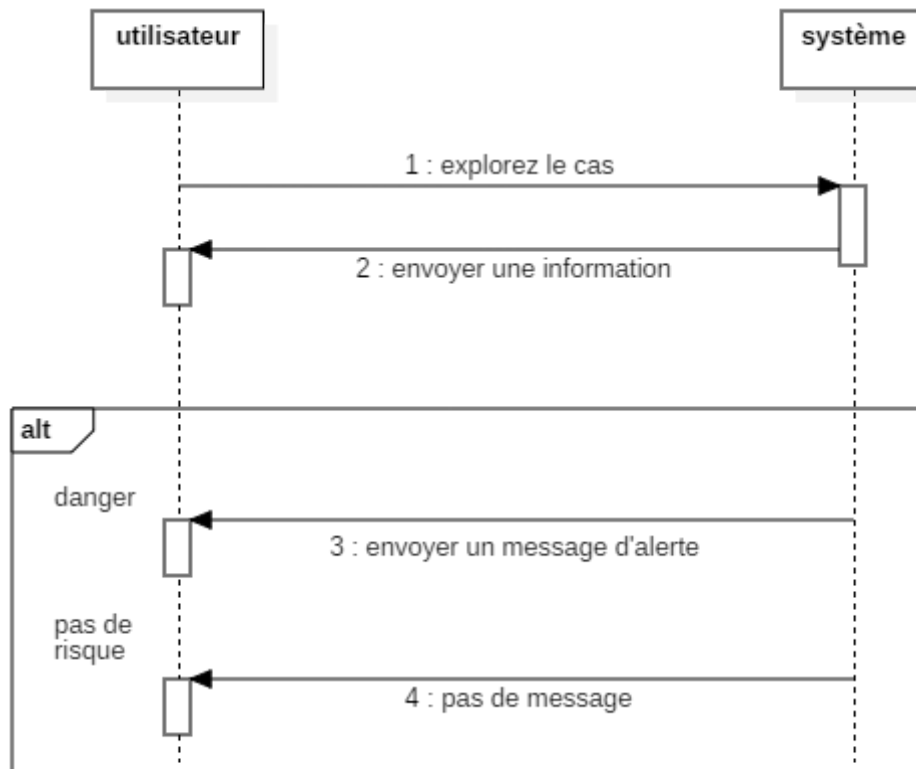


Figure 32 Diagramme de séquence utilisateur.

## Détecteurs

### Dans le cas d'un capteur de gaz :

- Détection du changement de type d'air

### Dans le cas du capteur de température :

- Détecter les changements de température.
- Alerte lorsqu'une étincelle électrique se produit.

### Dans le cas du capteur d'humidité :

- Détecter les changements d'humidité.

### Dans le cas du capteur de lumière :

- Détecter les changements d'éclairage.

### Dans le cas d'un capteur de mouvement :

- Détecter les changements de mouvement.

### Dans le cas d'un capteur des ports :

- Détecter l'ouverture des portes.

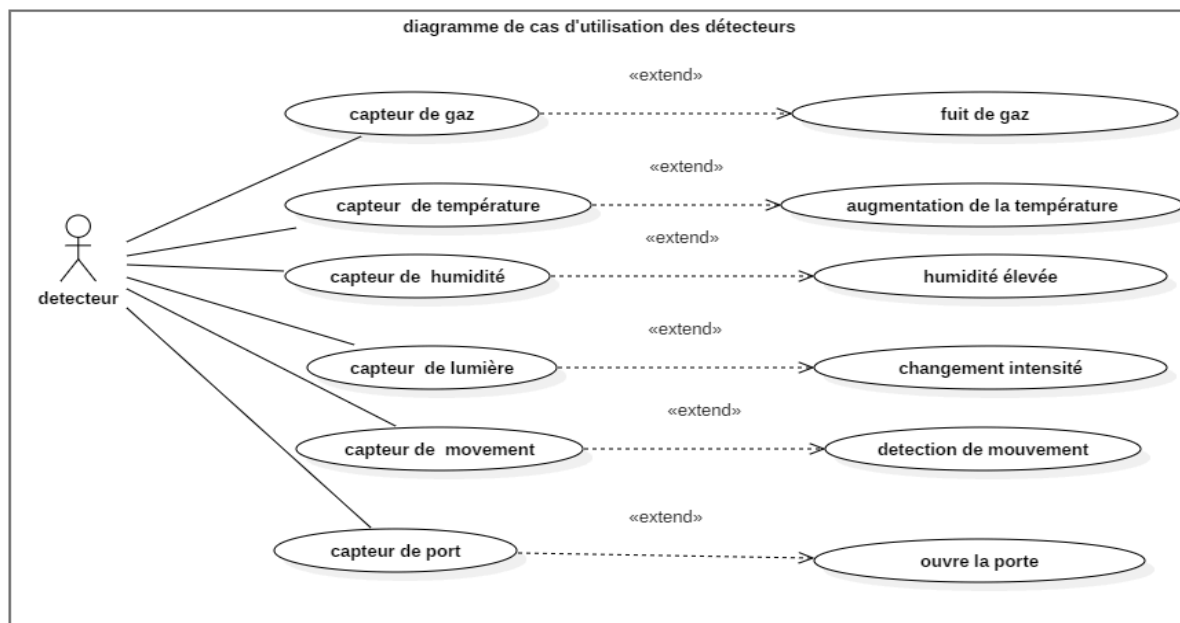


Figure 33 Représente le schéma de cas d'utilisation des détecteurs.

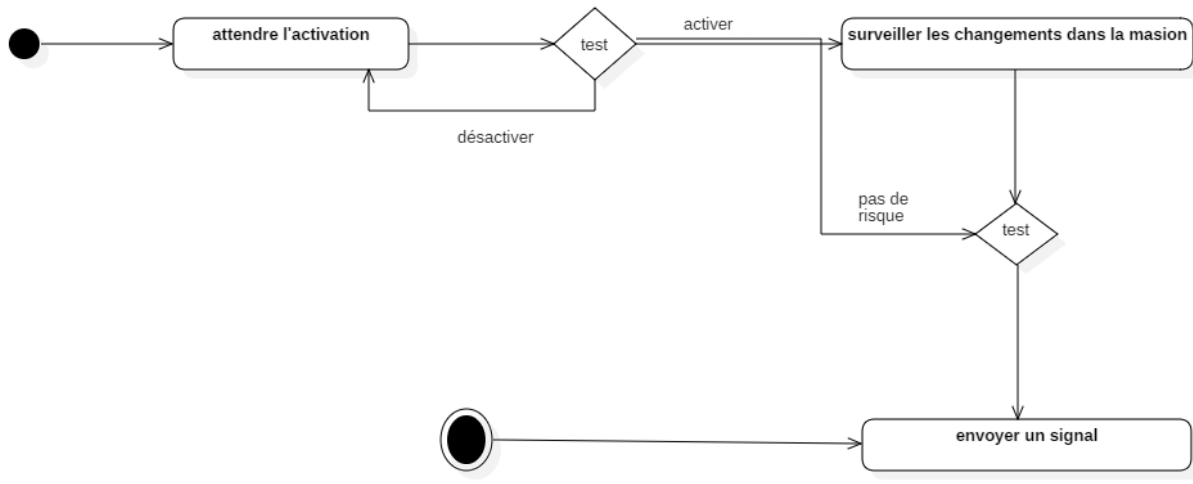


Figure 34 Représente le diagramme d'activité des détecteurs.

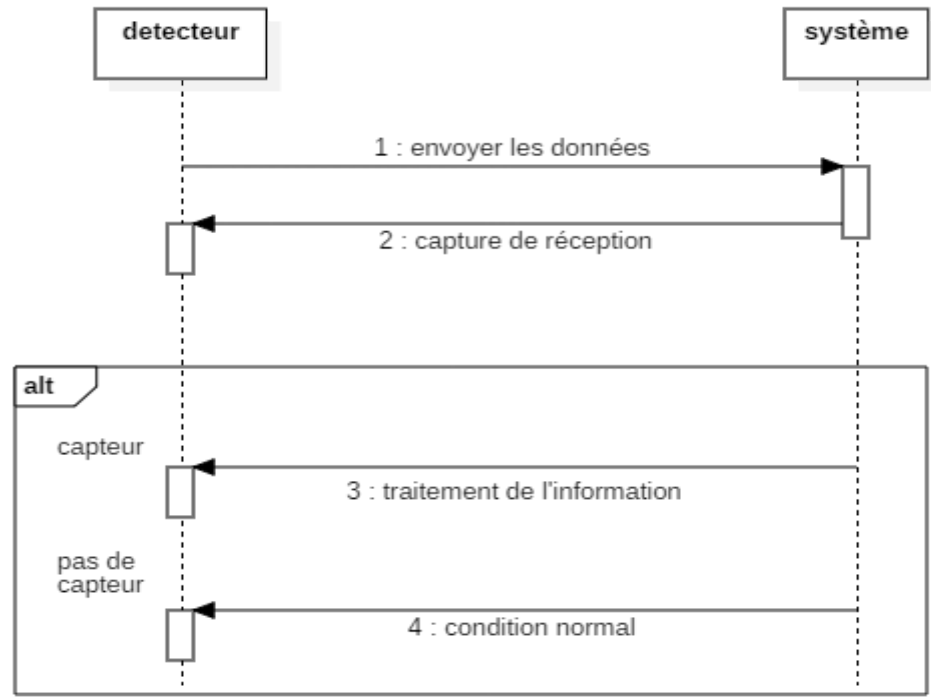


Figure 35 Représente le diagramme de séquence des détecteurs.

**L'actionneur**

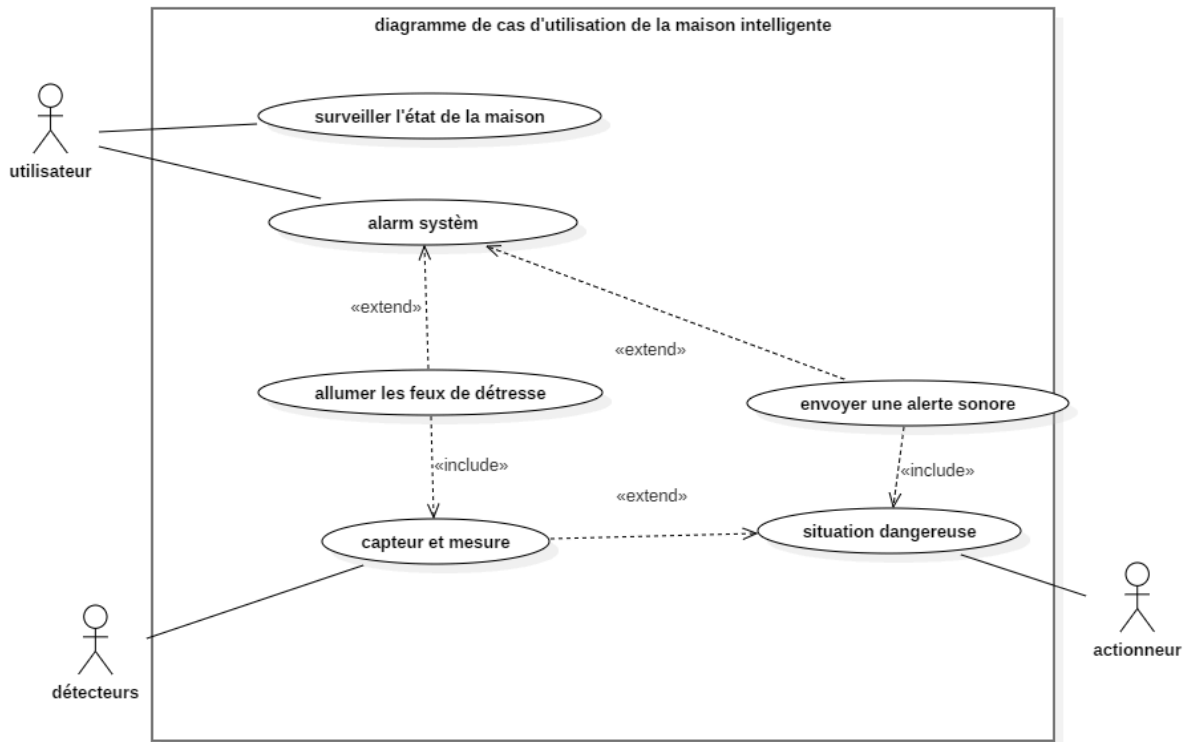


Figure 36 Représente le diagramme d'activité de l'actionneur.

La figure 37 représente le diagramme d'activité de l'actionneur

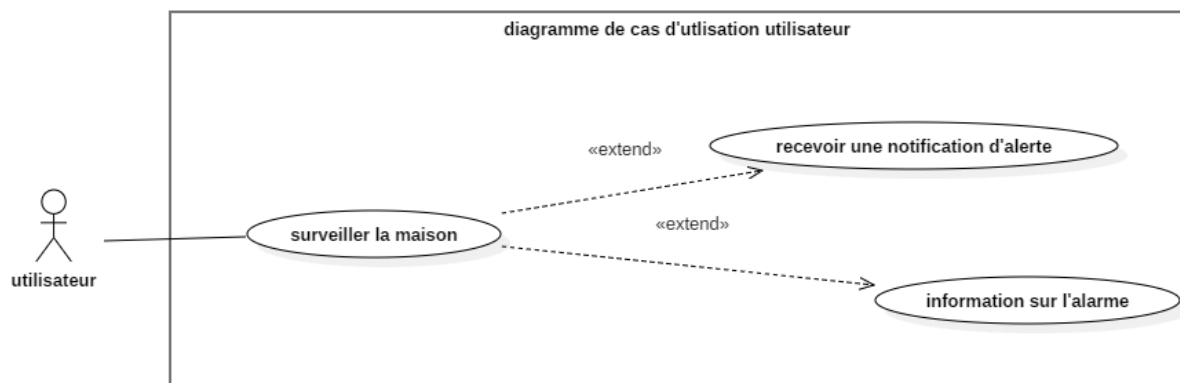


Figure 37 Diagramme d'activité de l'actionneur.

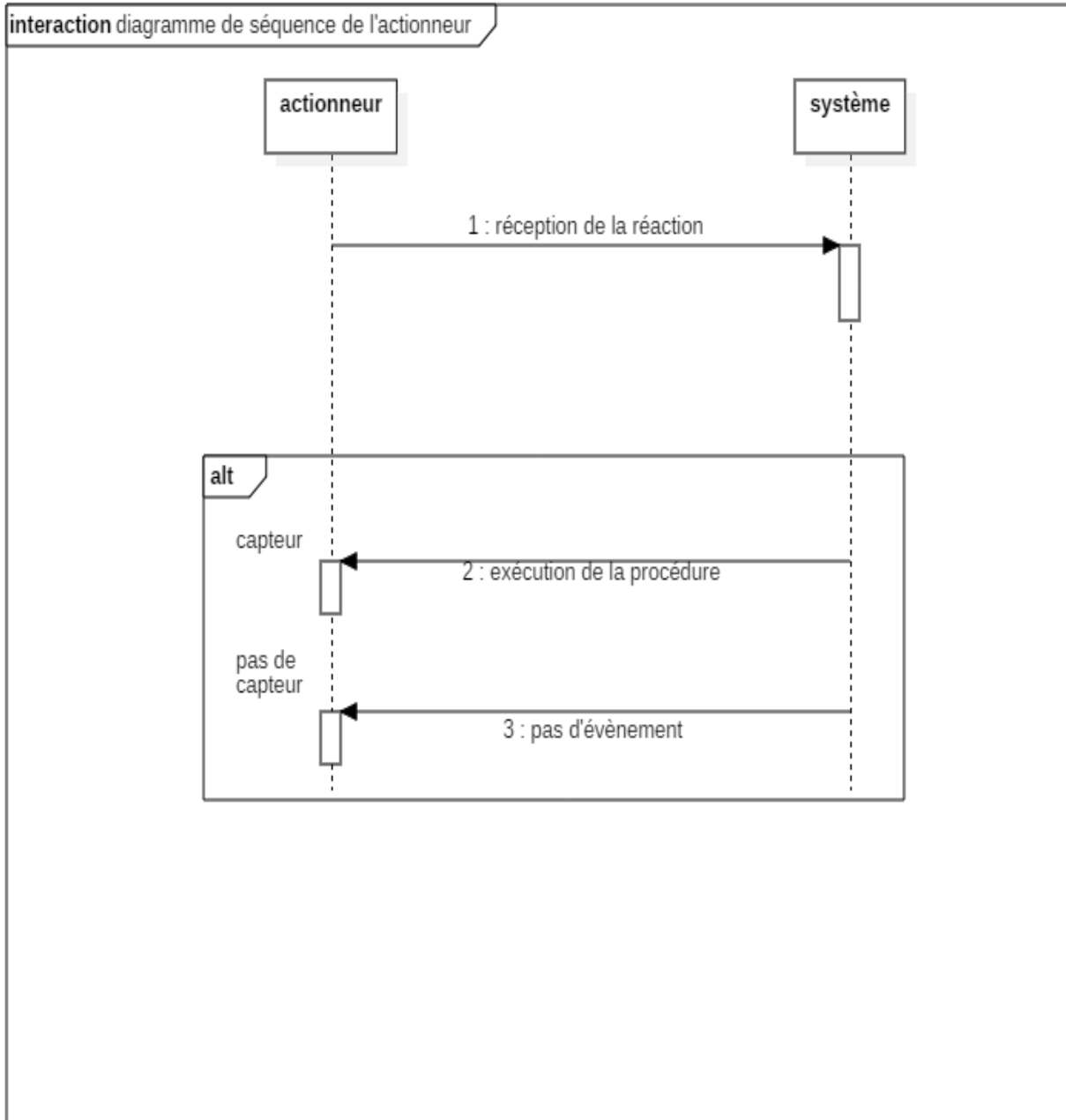


Figure 38 Diagramme de séquence d'actionneur.

## 4.4. Scénarios

### Scénario 1 -Détection de fuites de gaz dans les domiciles

Dans ce scénario, des capteurs de gaz sont installés dans des domiciles pour détecter les fuites potentielles de gaz combustible tel que le propane ou le méthane. Ces capteurs sont généralement placés près des appareils à gaz, tels que les cuisinières, les chauffe-eau et les systèmes de chauffage.

Les capteurs de gaz surveillent en continu la concentration de gaz dans l'air et transmettent les données collectées à une passerelle IdO domestique. Cette passerelle agrège les données de tous les capteurs de gaz dans la maison et les envoie à une plateforme dédiée à la gestion de la sécurité. Les données sont analysées en temps réel pour détecter toute augmentation anormale de la concentration de gaz, ce qui pourrait indiquer une fuite. En cas de détection de fuite, des alertes sont envoyées aux occupants de la maison via une application mobile ou un système de notification, leur permettant de prendre des mesures immédiates pour évacuer la zone et contacter les autorités compétentes.

### Scénario 2 - Sécurité intelligente avec vidéosurveillance

**Description :** Ce scénario utilise des caméras de vidéosurveillance pour surveiller la maison de l'intérieur et de l'extérieur et des capteurs installés sur les portes et fenêtres. Lorsqu'un mouvement ou d'autres événements suspects sont détectés, le système envoie des notifications aux propriétaires et enregistre des vidéos pour une analyse ultérieure.

**Comment ça marche :** Les caméras de sécurité sont stratégiquement placées à des endroits clés de la maison, comme l'entrée principale, le jardin et les zones sensibles comme le garage. Lorsqu'un mouvement est détecté, le système envoie immédiatement une notification aux propriétaires sur leurs appareils mobiles, leur permettant de visualiser en direct la caméra concernée. Dans le même temps, le système enregistre automatiquement une vidéo de l'événement suspect pour référence ultérieure.

De plus, si une activité inhabituelle est détectée alors que la maison est censée être vide, le système peut déclencher des alarmes, contacter les autorités locales et envoyer des notifications d'urgence aux propriétaires pour une action immédiate.

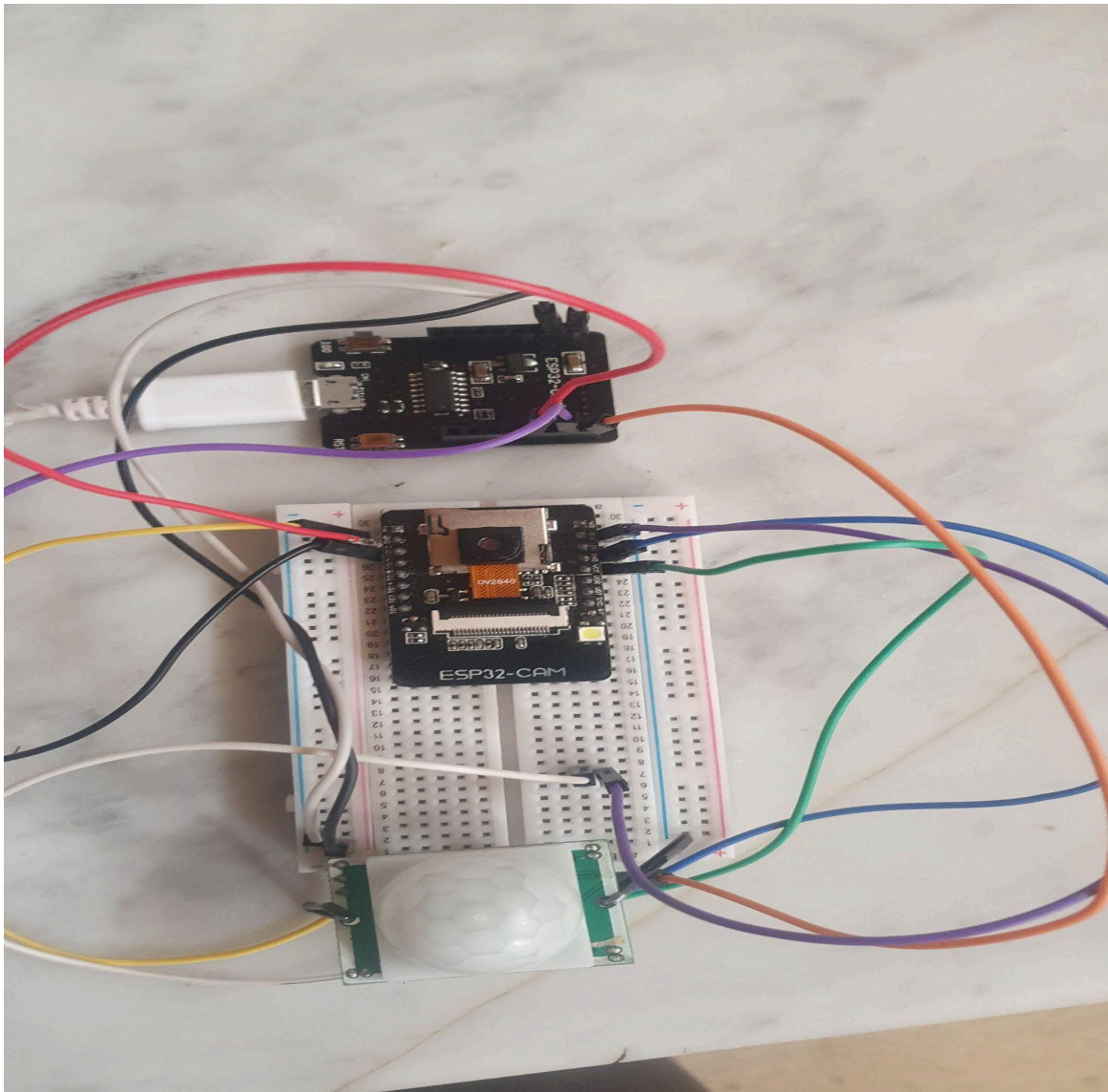


Figure 39 Dispositifs réels mis en œuvre dans notre expérience

## 4.5. Conclusion

Dans ce chapitre, nous avons couvert certains des détails importants qui constituent les piliers du projet. Nous avons proposé une conception et une architecture pour la surveillance et la surveillance de la maison intelligente.

Nous nous sommes assurés de réaliser des diagrammes de cas d'utilisation et des diagrammes d'activités pour le système utilisateur et diagramme de classes. La mise en œuvre de cette application sera détaillée lors de la présentation de notre travail.

# Chapitre 05 Conclusion générale

## 5.1. Contributions de projet

Le projet de maison intelligente que nous avons développé représente une avancée significative dans le domaine de l'automatisation domestique, en intégrant des technologies IdO avancées pour offrir une flexibilité et une personnalisation maximales.

En adoptant une approche basée sur des dispositifs séparés pour chaque appareil, nous avons simplifié l'installation et optimisé l'adaptabilité selon les besoins spécifiques des utilisateurs. L'intégration du protocole MQTT a été essentielle pour assurer une transmission de données fluide et fiable, garantissant ainsi une communication efficace entre les appareils et une gestion intelligente via des capteurs et actionneurs. Ce projet a non seulement enrichi notre expertise en automatisation intelligente, mais il a aussi renforcé notre méthode de pensée et notre capacité à développer des interfaces utilisateur simples et intuitives pour un contrôle complet du système.

## 5.2. Critique du Projet de Maison Intelligente

Le projet de maison intelligente que nous avons développé intègre des composants IdO, des actionneurs et des capteurs pour permettre une interaction intelligente non seulement entre les appareils, mais aussi avec les propriétaires. Cette intégration, facilitée par le protocole MQTT, a montré une grande flexibilité et compatibilité pour la personnalisation de la maison. Cependant, malgré les nombreux avantages et innovations apportés par ce projet, certaines critiques peuvent être formulées pour améliorer les futures versions.

**Complexité de l'Installation et de la Maintenance :** Bien que les dispositifs soient faciles à installer individuellement, la complexité globale du système peut poser des défis en termes de maintenance et de dépannage.

**Sécurité et Confidentialité :** Bien que la plateforme devrait être conçue pour être sécurisée, il est crucial de rester vigilant quant aux menaces de cybersécurité. Des audits réguliers de sécurité et des mises à jour fréquentes des logiciels de sécurité sont essentiels pour protéger les données des utilisateurs.

Le plus grand inconvénient est le manque flagrant de matériels réels pour expérimenter notre travail.

## 5.3. Travaux futurs et perspectives

L'amélioration de l'interface utilisateur pour rendre le système plus facile d'accès est un objectif majeur. Une interface plus intuitive, avec des options de commande vocale et d'excellents assistants, facilite l'utilisation de la maison intelligente pour les personnes.



## Références :

- [1] [https://www.giga-concept.fr/technologies/protocole-lorawan/#:~:text=LoRaWAN%20\(LoRa%20Wide%20Area%20Network,inscrit%20dans%20les%20technologies%20LPWAN](https://www.giga-concept.fr/technologies/protocole-lorawan/#:~:text=LoRaWAN%20(LoRa%20Wide%20Area%20Network,inscrit%20dans%20les%20technologies%20LPWAN).
- [2] <https://www.paessler.com/fr/it-explained/lpwa>.
- [3] <https://sendbird.com/fr/developer/tutorials/websocket-vs-http-communication-protocols>.
- [4] [https://aws.amazon.com/fr/what-is/mqtt/#:~:text=MQTT%20poss%C3%A8de%20des%20fonctionnalit%C3%A9s%20int%C3%A9gr%C3%A9es,exactement%20une%20fois%20\(2\)](https://aws.amazon.com/fr/what-is/mqtt/#:~:text=MQTT%20poss%C3%A8de%20des%20fonctionnalit%C3%A9s%20int%C3%A9gr%C3%A9es,exactement%20une%20fois%20(2)).
- [5] <https://cormind.com/lorawan>
- [6] <https://www.linkedin.com/pulse/lorawan-private-network-serious-alternative-iot-project-anciaux>
- [7] <https://www.instrumentchoice.com.au/news/what-is-sigfox>
- [9] <https://sendbird.com/fr/developer/tutorials/websocket-vs-http-communication-protocols>
- [8] <https://pressbooks.bccampus.ca/cellulariot/chapter/chapter-4/>
- [10] <https://iotbyhvm.ooo/mqtt-protocol-mqtt-in-depth-mqtt-vs-coap/>
- [11] <https://fr.mathworks.com/help/thingspeak/mqtt-basics.html>
- [12] <https://www.moneycontrol.com/news/technology/explainer-what-is-m2m-communication-and-what-are-its-applications-8889611.html>
- [13] <https://www.scaleway.com/fr/blog/decouverte-du-protocole-mqtt/>
- [14] <https://www.emqx.com/en/blog/mosquitto-mqtt-broker-pros-cons-tutorial-and-modern-alternatives>
- [15] <https://snapcraft.io/mosquitto>
- [16] <https://cedalo.com/blog/mqtt-on-android-guide-using-mosquitto/>
- [17] <https://edistechlab.com/mqtt-einfach-erklart/?v=fa3c7f2b5dae>
- [18] <https://www.lexhan-group.fr/blog/connectivite/plateforme-iot/#:~:text=Une%20plateforme%20IdO%20est%20un,une%20communication%20M2M%20entre%20eux>.
- [19] <https://www.data-bird.co/blog/langage-python>.
- [20] Srdjan Krco, Boris Pokric, and Francois Carrez. “Designing IdO Architecture(s): A European Perspective”. In: Internet of Things (WF-IdO), 2014 IEEE World Forum on. IEEE. 2014, pp. 79–84.
- [21] IdO6 - Universal Integration of the Internet of Things through an IPv6-based Service Oriented Architecture Enabling Heterogeneous Components Interoperability. <https://www.ietf.org/>. (last accessed : 01-04-2024).
- [22] Julien Mineraud et al. “A gap analysis of Internet-of-Things platforms”. In: Computer Communications 89 (2016), pp. 5–16.
- [23] <https://developer.mozilla.org/fr/docs/Web/HTML>.

- [24] Manveer Joshi and Bikrampal Kaur. Web integrated smart home infrastructure using internet of things. *Int. J. Eng. Res. Gen. Sci*, 3(6):153–158, 2015.
- [25] Moataz Soliman, Tobi Abiodun, Tarek Hamouda, Jiehan Zhou, and Chung-Horng Lung. Smart home: Integrating internet of things with web services and cloud computing. In *2013 IEEE 5th international conference on cloud computing technology and science*, volume 2, pages 317–320. IEEE, 2013.
- [26] Xiaoshan Li, Zhiming Liu, and He Jifeng. A formal semantics of uml sequence diagram. In *2004 Australian Software Engineering Conference. Proceedings.*, pages 168–177. IEEE, 2004
- [27] Ricardo Melo Bastos and Duncan Dubugras A Ruiz. Extending uml activity diagram for workflow modeling in production systems. In *Proceedings of the 35th Annual Hawaii International Conference on System Sciences*, pages 3786–3795. IEEE, 2002
- [28] Daniela Berardi, Diego Calvanese, and Giuseppe De Giacomo. Reasoning on uml class diagrams. *Artificial intelligence*, 168(1-2):70–118, 2005.
- [29] Hans-Erik Eriksson, Magnus Penker, Brian Lyons, and David Fado. *UML 2 toolkit*. John Wiley & Sons, 2003.
- [30] <https://fr.wikipedia.org/wiki/Sigfox>.
- [31] <https://christophe-f.developpez.com/tutoriels/css/cours-css/introduction-css/>.
- [32] <https://developer.mozilla.org/fr/docs/Web/JavaScript/Guide/Introduction>.
- [33] [https://github.com/shamblett/mqtt\\_client](https://github.com/shamblett/mqtt_client).
- [34] [https://aws.amazon.com/fr/what-is/mqtt/#:~:text=MQTT%20poss%C3%A8de%20des%20fonctionnalit%C3%A9s%20int%C3%A9gr%C3%A9es,exactement%20une%20fois%20\(2\)](https://aws.amazon.com/fr/what-is/mqtt/#:~:text=MQTT%20poss%C3%A8de%20des%20fonctionnalit%C3%A9s%20int%C3%A9gr%C3%A9es,exactement%20une%20fois%20(2).).
- [35] <https://www.talsom.com/insights/definition-internet-des-objets/>.
- [36] <https://www.digora.com/fr/blog/quest-ce-que-iiot-et-pourquoi-mener-une-strategie-iiot>.
- [37] <https://docplayer.fr/86884324-Memoire-de-master-recherche-authentification-dans-l-iiot.html>.
- [38] [https://www.freertos.org/mqtt/index.html#:~:text=or%20small%20MPU\)-,Introduction,Things%20\(IoT\)%20use%20cases](https://www.freertos.org/mqtt/index.html#:~:text=or%20small%20MPU)-,Introduction,Things%20(IoT)%20use%20cases).
- [39] <https://www.eprojet.fr/cours/pdf/bootstrap-support.pdf>.
- [40] [https://www.researchgate.net/figure/Smart-Home-http-smarthomeenergycouk\\_fig1\\_325318697](https://www.researchgate.net/figure/Smart-Home-http-smarthomeenergycouk_fig1_325318697)
- [41] <https://link.springer.com/article/10.1007/s11235-021-00787-w>
- [42] <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0231615>