



**REPUBLIQUE TUNISIENNE**

\*\*\*\*\*

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR, DE LA  
RECHERCHE SCIENTIFIQUE ET DE LA TECHNOLOGIE

\*\*\*\*\*

DIRECTION GENERALE DES ETUDES TECHNOLOGIQUES

\*\*\*\*\*

INSTITUT SUPERIEUR DES ETUDES TECHNOLOGIQUES DE  
CHARGUIA

\*\*\*\*\*

**Département Technologies de l'Informatique**



**RAPPORT DE PROJET DE FIN D'ETUDES**

Elaboré en vue de l'obtention du diplôme de

**Licence Appliquée en Technologies de l'Informatique**

Parcours : Systèmes Embarqués & Mobiles

**Sujet :**

**Conception et réalisation d'une solution domotique**  
**Smart Home « DARINET »**

**Elaboré par :**

BOUSSETTA Roua

&

BOUGUERRA Ons

**Encadrant professionnel :**

Mr. Touati Selim

**Encadrant pédagogique :**

Mr. Lamari Mounir

**Société d'accueil : Exsol Systems**

**Année Universitaire : 2019/2020**

# Dédicaces

Je dédie ce travail à :

Ma mère, la source de tendresse et l'exemple de dévouement qui n'a pas cessé de m'encourager. Aucune dédicace ne saurait être assez éloquente pour exprimer ce que vous méritez pour tous vos efforts et sacrifices.

Mon père, aucune dédicace ne saurait exprimer l'amour, l'estime le dévouement et le respect que j'ai toujours eu pour vous. Ce travail est le fruit de vos sacrifices consentis pour mon éducation.

Mes cousins et mes cousines qui m'ont soutenu et m'ont encouragé.

Mes amis et mes camarades qui ont transformé mon parcours scolaire en une extraordinaire aventure.

Tous mes professeurs, veuillez trouver dans ce modeste travail ma reconnaissance pour tous vos efforts.



Roua BOUSSETTA

Je dédie ce travail :

À mes parents pour l'amour, l'affection et la bienveillance dont ils m'entouraient, pour leurs efforts et leurs sacrifices qu'ils ont dû faire, pour leurs patiences et leurs encouragements.

Pour mon cher frère Houssein et ma chère sœur pour leurs encouragements incessants.

Tous mes amis, pour toute personne qui a participé de près ou de loin à la l'élaboration de ce projet.



Ons BOUGUERRA

# Remerciements

Nous aimerons bien profiter de cette occasion pour exprimer notre sincère gratitude et remerciement pour tous ceux qui ont contribué d'une façon directe ou indirecte à la réussite et au bon déroulement de ce stage.

Nous profitons de bien remercier « EXSOL SYSTEMS » de nous avoir acceptées et de nous avoir affectées à un savoureux et intéressant projet.

Un grand merci s'adresse également à notre encadrant professionnel 'Mr. Touati Selim' de nous avoir guidées, critiquées et conseillées tout au long du projet.

Un profond remerciement et une sincère gratitude et considération s'adresse à 'Mr. Lamari Mounir' pour ses précieux efforts et ses encouragements qui nous ont toujours poussées à faire de notre mieux et qui ont le grand impact dans la réussite du ce projet.

Nous voudrions exprimer également, nos remerciements au directeur et au chef de département de notre établissement 'ISET CHARGUIA'.

Enfin, nous manifestons notre profond respect aux membres de jury d'avoir accepté d'évaluer notre travail. Nous espérons être dignes aux qualités que vous exceptez.



# Table des matières

<b>Introduction générale</b> .....	1
<b>Chapitre I : Présentation de l'organisme d'accueil</b> .....	1
<b>Introduction</b> .....	2
<b>I. Présentation de l'entreprise</b> .....	2
<b>II. Domaine d'action de l'entreprise</b> .....	2
<b>III. Organigramme de l'entreprise</b> .....	2
<b>Conclusion</b> .....	3
<b>Chapitre II: Présentation de cadre du projet</b> .....	2
<b>Introduction</b> .....	4
<b>I.Problématique</b> .....	4
<b>II.Etude de l'existant : description et critique</b> .....	4
<b>1.Analyse de la solution JEEDOM</b> .....	5
<b>a.Description</b> .....	5
<b>b.Avantages</b> .....	5
<b>c.Critique</b> .....	5
<b>2. Analyse de la solution Domoticz</b> .....	6
<b>a.Description</b> .....	6
<b>b.Avantages</b> .....	6
<b>c.Critiques</b> .....	6
<b>3. Analyse de la solution Home Assistant</b> .....	7
<b>a.Description</b> .....	7
<b>b.Avantages</b> .....	7
<b>c.Critiques</b> .....	7
<b>4. Analyse de la solution OpenHab</b> .....	8
<b>a.Description</b> .....	8
<b>b.Avantages</b> .....	8
<b>c.Critiques</b> .....	9
<b>5. Analyse de la solution OpenRemote</b> .....	9
<b>a.Description</b> .....	9
<b>b.Avantages</b> .....	9
<b>c.Critiques</b> .....	9
<b>Synthèse</b> .....	9
<b>III.Solution proposée</b> .....	10

IV.Méthodologies de travail .....	10
Conclusion.....	12
<i>Chapitre III : Analyse et spécification des besoins</i> .....	4
Introduction .....	13
I.Description générale .....	13
1.Objectif de l'application.....	13
2.Fonctionnement de l'application.....	13
II.Etude des besoins.....	13
1.Analyse des besoins fonctionnels.....	13
2.Analyse des besoins techniques .....	14
III.Diagramme de cas d'utilisation .....	14
1.Identification des acteurs .....	14
2.Diagramme de cas d'utilisation général .....	15
IV.Etude matérielle .....	15
1.Etude des capteurs.....	16
2.Etude des actionneurs .....	19
3.Etude des cartes .....	20
Conclusion.....	22
<i>Chapitre IV: Conception</i> .....	23
Introduction .....	23
I.Architecture globale de la solution.....	23
1.Architecture matérielle .....	24
2.Architecture logicielle .....	24
II.Vue statique .....	26
1.Diagramme de cas d'utilisation.....	26
1.1.Diagramme de cas d'utilisation : Gestion des éclairages .....	26
1.1.1.Description textuelle.....	26
a.Allumer les lampes.....	26
b.Eteindre les lampes.....	27
1.2.Diagramme de cas d'utilisation : Gestion de sécurité .....	27
1.2.1.Description textuelle.....	28
a.Activer alarme.....	28
b.Désactiver alarme .....	28
1.3.Diagramme de cas d'utilisation : Gestion des portes .....	29
1.3.1.Description textuelle.....	29
a.Ouvrir la porte.....	29

b.Fermer la porte.....	30
1.4.Diagramme de cas d'utilisation : Gestion des volets .....	30
1.4.1.Description textuelle.....	31
a.Ouvrir le volet.....	31
b.Fermer le volet.....	31
1.5.Diagramme de cas d'utilisation : Gestion d'aération.....	32
1.5.1.Description textuelle.....	32
a.Mettre en marche le ventilateur.....	32
b.Mettre en arrêt le ventilateur.....	33
1.6.Diagramme de cas d'utilisation : Consultation des informations .....	33
1.6.1.Description textuelle.....	34
a.Créer un compte .....	34
b.Accepter les conditions de l'application .....	35
c.S'authentifier .....	36
2.Diagramme de déploiement.....	36
3.Diagramme de composants.....	37
III.Vue dynamique .....	39
1.Diagramme de séquence.....	39
1.1.Application Mobile .....	39
a.Diagramme de séquence relatif au « Consultation des informations ».....	39
1.2.Application Web .....	40
1.3.Système automatique.....	41
2.Diagramme de classe .....	41
3.Diagramme d'activité.....	42
Conclusion.....	44
Chapitre V : <i>Réalisation</i> .....	45
Introduction .....	45
I.Environnement de travail .....	45
1.Environnement matériel .....	45
1.1.Matériel de développement.....	45
1.2.Matériel du nœud capteurs/actionneurs.....	46
1.2.1.Arduino UNO.....	46
a.Description de la carte Arduino UNO.....	46
b.Spécifications techniques de la carte Arduino UNO .....	46
1.2.2.Capteur dht11 .....	47
a.Description.....	47

b.Spécifications techniques du capteur dht11 .....	47
c.Branchement du capteur dht11 .....	47
1.2.3.Capteur détecteur de mouvement PIR .....	48
a.Description.....	48
b.Spécifications techniques du capteur PIR .....	48
c.Branchement du capteur .....	49
d.Réglage de la sensibilité du capteur .....	49
1.2.4.Capteur niveau d'eau .....	49
a.Description.....	49
b.Spécifications techniques .....	49
c.Branchement .....	50
1.2.5.Capteur MQ135 .....	50
a.Description.....	50
b.Spécifications techniques .....	50
c.Branchement du capteur MQ135 .....	51
1.2.6.Capteur RFID .....	51
a.Description.....	51
b.Spécifications techniques .....	52
c.Branchement .....	52
1.2.7.Servomoteur .....	52
a.Description.....	52
b.Spécifications techniques .....	53
c.Branchement du servomoteur.....	53
1.2.8.Buzzer .....	54
a.Description.....	54
b.Spécifications techniques .....	54
c.Branchement .....	54
1.2.9.Ventilateur.....	54
a.Description.....	54
b.Spécifications techniques .....	54
c.Branchement .....	55
1.3.Matériel du Gateway .....	55
a.Description de la carte Raspberry pi 3 B+ .....	55
b.Spécifications techniques de la carte Raspberry pi 3 B+ .....	56
1.4.Matériel de communication .....	56
a.Description du module ESP8266.....	56



b.Spécifications techniques .....	56
c.Branchement .....	56
2.Environnement logiciel .....	57
2.1.Application embarquée .....	57
a.Système d'exploitation .....	57
b.Logiciel de formatage de la carte SD .....	57
c.Logiciel de transfert.....	58
d.Logiciel de scan IP.....	58
e.Logiciel PuTTY .....	58
f.Logiciel OpenHab .....	59
2.2.Application mobile.....	59
a.Environnement de développement.....	59
2.3.Application Web .....	59
a.Environnement de développement .....	59
b.Langages de développement .....	60
2.4.Logiciel de modélisation .....	60
a.StarUML.....	60
2.5.Logiciel Sweet Home 3D.....	61
II.Architecture système.....	61
III.Travail réalisé : Partie software .....	62
1.Application Mobile .....	62
1.1.Acceptation des conditions de l'application .....	62
1.2.Création du compte .....	62
1.3.Authentification .....	63
1.4.Interface de consultation.....	63
2.Application web .....	64
IV.Travail réalisé : Partie hardware .....	65
1.Câblage du nœud de capteurs/actionneurs .....	65
2.Maquette.....	67
Conclusion et perspectives .....	69
Nétographie	
Bibliographie	
Annexes	

# Liste des figures

Figure 1 : Logo de l'entreprise "EXSOL SYSTEMS" .....	2
Figure 2 : Organigramme de l'entreprise "EXSOL SYSTEMS" .....	3
Figure 3 : Types d'accidents domestiques .....	4
Figure 4 : Interface Jeedom .....	5
Figure 5 : Interface Domoticz .....	6
Figure 6 : Interface Home Assistant .....	7
Figure 7 : Interface OpenHab .....	8
Figure 8 : Interface OpenRemote .....	9
Figure 9 : Diagramme des cas d'utilisation générale du système .....	15
Figure 10 : Architecture globale de la solution .....	23
Figure 11 : Schéma d'architecture matérielle .....	24
Figure 12 : Schéma d'architecture logicielle .....	24
Figure 13 : Diagramme cas d'utilisation « Gestion d'éclairage » .....	26
Figure 14: Diagramme de cas d'utilisation « Gestion de sécurité » .....	27
Figure 15 : Diagramme de cas d'utilisation « Gestion des portes » .....	29
Figure 16 : Diagramme de cas d'utilisation « Gestion des volets » .....	30
Figure 17 : Diagramme de cas d'utilisation « Gestion d'aération » .....	32
Figure 18 : Diagramme de cas d'utilisation « Consultation des informations » .....	33
Figure 19 : Diagramme de déploiement .....	37
Figure 20 : Diagramme de composant .....	38
Figure 21 : Diagramme de séquence de l'application Mobile : cas d'utilisation « Consultation des informations » .....	40
Figure 22 : Diagramme de séquence de l'application Web .....	41
Figure 23 : Diagramme de classe .....	42
Figure 24 : Diagramme d'activité : « Consultation des informations » .....	43
Figure 25 : Matériel du nœud capteurs/actionneurs .....	46
Figure 26 : Schéma de branchement du capteur dht11 .....	47
Figure 27 : Fonctionnement du capteur PIR .....	48
Figure 28 : Branchement du capteur PIR .....	49
Figure 29 : Le réglage de la sensibilité du capteur PIR .....	49
Figure 30 : Branchement du capteur niveau d'eau ST045 .....	50
Figure 31 : Branchement du capteur gaz MQ135 .....	51
Figure 32: schéma de branchement du module RFID .....	52
Figure 33 : Branchement du servomoteur .....	53
Figure 34 : Branchement du Buzzer .....	54
Figure 35 : Branchement du ventilateur .....	55
Figure 36 : Matériel du Gateway .....	55
Figure 37 : Branchement du module ESP8266 .....	56
Figure 38 : Logo de système d'exploitation "Raspbian" .....	57
Figure 39 : Logo de logiciel de formatage de la carte SD "SDformatter" .....	57

<b>Figure 40 : Logo de logiciel de transfert SE "Win32 disk imager"</b> .....	58
<b>Figure 41 : Logo de logiciel de scan IP "Advanced IP scanner"</b> .....	58
<b>Figure 42 : Logo de logiciel "PuTTY"</b> .....	58
<b>Figure 43 : Logo de logiciel "OpenHab"</b> .....	59
<b>Figure 44 : Logo de l'environnement de développement "Android Studio"</b> .....	59
<b>Figure 45 : Logo de l'environnement de développement "Visual Studio Code"</b> .....	59
<b>Figure 46 : logo PHP</b> .....	60
<b>Figure 47 : logo java</b> .....	60
<b>Figure 48 : Logo de logiciel de modélisation "StarUML"</b> .....	60
<b>Figure 49 : logo de logiciel Sweet Home 3D</b> .....	61
<b>Figure 50 : Architecture système</b> .....	61
<b>Figure 51 : Interface d'acceptation des conditions de l'application mobile</b> .....	62
<b>Figure 52 : Interface d'inscription</b> .....	63
<b>Figure 53 : Interface d'authentification</b> .....	63
<b>Figure 54 : Interface de consultation des informations</b> .....	64
<b>Figure 55 : schéma de câblage du nœud de capteurs/actionneurs</b> .....	66
<b>Figure 56 : Maquette 2D</b> .....	67
<b>Figure 57 : Maquette 3D</b> .....	67
<b>Figure 58 : installation système d'exploitation Raspbian</b> .....	76
<b>Figure 59 : formatage de la carte SD</b> .....	77

# Liste des tableaux

<b>Tableau 1 : Tableau comparatif des capteurs de température &amp; humidité</b>	16
<b>Tableau 2 : Tableau comparatif des capteurs détecteurs de mouvement</b>	17
<b>Tableau 3 : Tableau comparatif des capteurs de gaz</b>	18
<b>Tableau 4 : Tableau comparatif des capteurs détecteurs de niveau d'eau</b>	19
<b>Tableau 5 : Tableau comparatif des différents moteurs</b>	20
<b>Tableau 6 : Tableau d'étude sur les cartes électroniques</b>	21
<b>Tableau 7 : Description textuelle de cas d'utilisation « Allumage des lampes »</b>	26
<b>Tableau 8 : Description textuelle de cas d'utilisation « Extinction des lampes »</b>	27
<b>Tableau 9 : Description textuelle de cas d'utilisation « Activer alarme »</b>	28
<b>Tableau 10 : Description textuelle de cas d'utilisation « Désactiver alarme »</b>	28
<b>Tableau 11 : Description textuelle de cas d'utilisation « Ouvrir la porte »</b>	29
<b>Tableau 12 : Description textuelle de cas d'utilisation « Fermer la porte »</b>	30
<b>Tableau 13 : Description textuelle de cas d'utilisation « Ouvrir le volet »</b>	31
<b>Tableau 14 : Description textuelle de cas d'utilisation « Fermer le volet »</b>	31
<b>Tableau 15 : Description textuelle de cas d'utilisation « Mettre en marche le ventilateur »</b>	32
<b>Tableau 16 : Description textuelle de cas d'utilisation « Mettre en arrêt le ventilateur »</b>	33
<b>Tableau 17 : Description textuelle de cas d'utilisation « Création du compte »</b>	34
<b>Tableau 18 : Description textuelle de cas d'utilisation « Accepter les conditions de l'application »</b>	35
<b>Tableau 19 : Description textuelle de cas d'utilisation « Authentification »</b>	36

# Liste des abréviations

2TUP : 2 tracks unified process

API : Application Programmable Interface

UML : Unified Modeling Language

IDE : Integrated Development Environment

JSON : JavaScript Object Notation

HTML : Hyper Text Markup Language

MQTT : Message Queuing Telemetry Transport

TCP : Transmission Control Protocol

IP : Internet Protocol

PIR : Passive Infrared Radar

USB : Universal Serial Bus

ICSP : Indian Centre for Space Physics

SPI : Serial Peripheral Interface

TWI : Two Wire Interface

Mbps : Mégabits par seconde

# Introduction générale

Dans le cadre de notre formation en systèmes embarqués et mobiles à ISET Charguia, et afin d'obtenir notre diplôme national de licence appliquée en technologies d'informatique, nous avons eu la chance d'effectuer notre projet au sein de l'entreprise « EXSOL SYSTEMS ».

Ce projet aura comme sujet « Conception et réalisation d'un système connecté » qui permet le contrôle environnemental des maisons à partir d'une application mobile ou encore à partir d'une application web.

En effet, le présent rapport, résultat de cette mission, s'articule autour de cinq grands axes divisés comme suit :

- Le premier chapitre présente l'organisme d'accueil : Nous présentons en premier lieu l'entreprise. En deuxième lieu, nous présentons son domaine d'action. En troisième lieu, nous décrivons son organigramme,
- Le deuxième chapitre présente le cadre du projet : Nous nous référons en premier lieu à la problématique traitée pour introduire par la suite notre projet. En deuxième lieu, nous décrivons et critiquons l'existant afin de donner en troisième lieu notre solution proposée. Enfin, nous précisons la méthodologie de conception adoptée,
- Le troisième chapitre est consacré à présenter et expliquer globalement ce que l'application est censée faire et ceci à travers la spécification des différents besoins fonctionnels et besoins techniques. Après, nous nous étudions les matérielles qui satisfont nos besoins,
- Le quatrième chapitre nommé « Conception » renferme l'architecture de notre solution ainsi que l'ensemble des diagrammes de modélisation que nous avons choisis pour mettre en évidence notre solution conceptuelle et détaillée,
- Le cinquième et dernier chapitre comporte une description détaillée des principales étapes de réalisation et implémentation de ce projet.

***Chapitre I :***  
***Présentation de l'organisme***  
***d'accueil***

## **Introduction**

Ce premier chapitre présente l'entreprise où nous avons effectué notre stage de quatre mois, son domaine d'action ainsi que son organigramme.

### **I. Présentation de l'entreprise**



**Figure 1 : Logo de l'entreprise "EXSOL SYSTEMS"**

Exsol Systems [1] est une nouvelle entreprise qui été lancée le 27 juillet 2018, elle propose des solutions créatives à de nombreux domaines d'activités.

### **II. Domaine d'action de l'entreprise**

Le domaine d'activité de Exsol Systems est très vaste et il peut être réduit à deux domaines, le premier est la réalité virtuelle, qui aide les architectes à donner aux clients une meilleure compréhension des plans lors du passage du plan 2D à la réalité virtuelle. Le second est le domaine des objets connectés. Comme elle possède d'une troisième activité ajoutée récemment formation professionnelle pour les entreprises

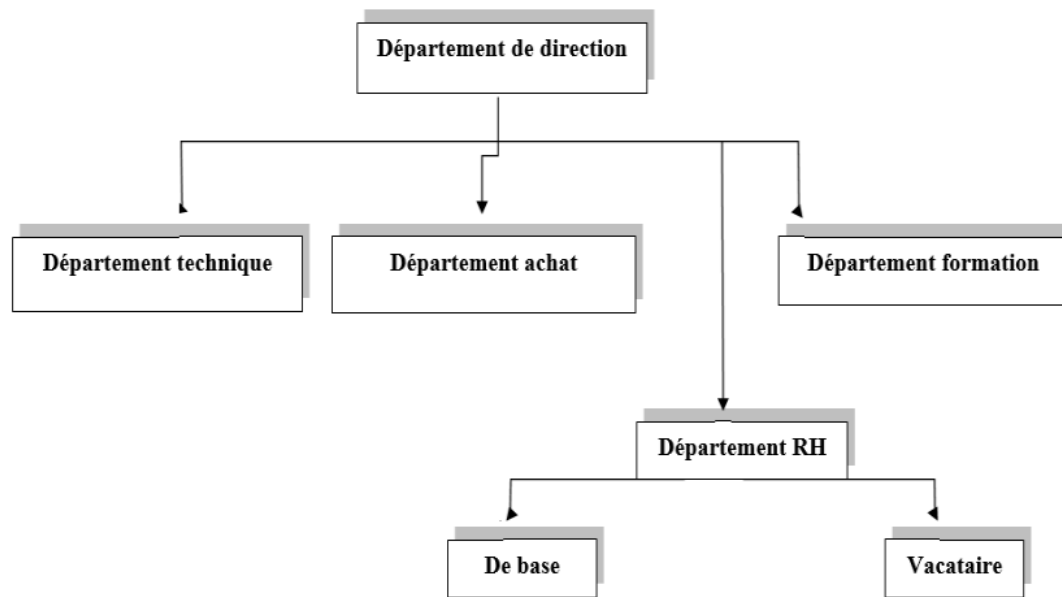
### **III. Organigramme de l'entreprise**

Du point de vue organisationnel, Exsol Systems est scindée en cinq départements comme l'indique la figure ci-dessous :

---

[1]Site officiel de la startup : <http://exsolsystems.tn/>





**Figure 2 : Organigramme de l'entreprise "EXSOL SYSTEMS"**

## **Conclusion**

Après avoir présenté l'organisme d'accueil et son domaine d'activité, on va entamer le deuxième chapitre là où nous avons présenté notre cadre de projet dans l'entreprise.

***Chapitre II:***  
***Présentation de cadre***  
***du projet***

## **Introduction**

Ce deuxième chapitre présente une démarche stratégique qui nous permettra d'avoir une vision globale assurant le bon déroulement de notre projet de fin d'études. En premier lieu, nous commençons par la problématique traitée, ensuite nous étudions l'existant afin de présenter notre solution proposée. Enfin, nous décrivons la méthodologie du travail adoptée.

### **I. Problématique**

Chaque année, les accidents de la vie courante touchent des millions de personnes et en tuent des milles. Parmi ces accidents, nous citons les accidents domestiques, avec toute sorte d'incendie, les fuites des gaz, dont les enfants et les seniors sont les premières victimes.

En outre, toutes les types de vols faits en absence des habitats ou encore en présence des enfants, Pourtant, beaucoup sont évitables si on adopte de bons réflexes.



**Figure 3 : Types d'accidents domestiques**

### **II. Etude de l'existant : description et critique**

Dans cette partie nous allons faire une analyse de quelques plateformes marchandes pour dégager les avantages et les inconvénients de chacune afin de pouvoir déterminer les besoins du client. Ensuite, nous formulerons une solution de la problématique en se basant sur les lacunes des existants.

## 1. Analyse de la solution JEEDOM

### a. Description

La solution JEEDOM [2] lancée en 2017, elle permet de fournir la sécurité, le confort et la meilleure gestion et suivi des énergies.

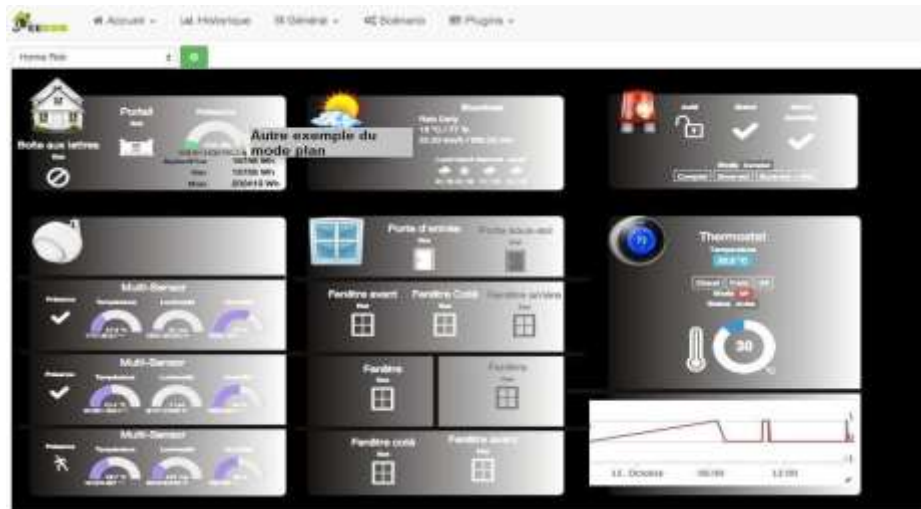


Figure 4 : Interface Jeedom

### b. Avantages

Parmi les avantages de cette solution nous citons :

- **Plateforme open source** : Le logiciel Jeedom est Open Source, permet un accès total au logiciel qui gère la domotique.
- **Autonome, simple et efficace** : Jeedom ne nécessite pas l'accès à des serveurs extérieurs pour fonctionner. Toute l'installation se gère en local et seulement le concernés peuvent y avoir accès pour garantir la sécurité.

### c. Critique

Parmi les inconvénients de cette solution nous citons :

- Complexité d'installation
- Certains plugins sont gratuits et d'autres payants

---

[2] : <https://www.jeedom.com/site/fr/>

## 2. Analyse de la solution Domoticz

### a. Description

Domoticz [3] une solution domotique lancée en 2012, elle permet de surveiller et de configurer des divers dispositifs tels que les éclairages, les commutateurs, divers capteurs tels que la température, la pluie et bien plus.



Figure 5 : Interface Domoticz

### b. Avantages

Parmi les avantages de la solution Domoticz nous citons :

- **Plateforme open source** : Ce système est conçu pour fonctionner sous différents systèmes d'exploitation.
- Compatible avec tous les navigateurs.
- Simple

### c. Critiques

Parmi les inconvénients de cette solution nous citons :

- L'interface de programmation simplifiée contient des bugs
- Pas de sauvegarde automatique intégrée
- Problème après les mises à jour.

---

[3] : <http://www.domoticz.com>

### 3. Analyse de la solution Home Assistant

#### a. Description

Home Assistant [4] une solution qui était lancée en 2012, elle peut être exécuté sur différents systèmes d'exploitation et qui offre la possibilité de la manipulation et l'automatisation des appareils.

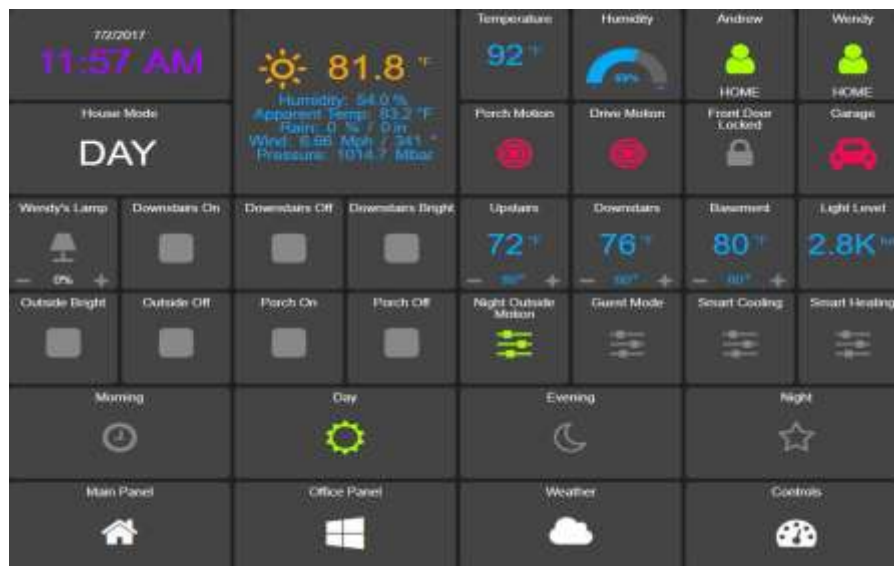


Figure 6 : Interface Home Assistant

#### b. Avantages

Parmi les avantages de la solution Home Assistant nous citons :

- **Plateforme open source** : Ce système est conçu pour fonctionner sous différents systèmes d'exploitation.
- Compatible avec tous les navigateurs.

#### c. Critiques

Parmi les inconvénients de cette solution nous citons :

- Problèmes de configuration
- Problèmes dans les dernières versions de système d'exploitation « hassbian ».

---

[4] : <https://www.home-assistant.io>

## 4. Analyse de la solution OpenHab

### a. Description

OpenHab [5] un logiciel open source lancé en 2016, développé avec le langage java et permettant d'effectuer des applications domotiques, il peut être installé sur un Raspberry pi.



Figure 7 : Interface OpenHab

### b. Avantages

Parmi les avantages de la solution OpenHab nous citons :

- **Plateforme open source** : Ce système est conçu pour fonctionner sous différents systèmes d'exploitation.
- **La flexibilité** : OpenHab offre l'outil le plus flexible disponible pour réaliser presque tous les souhaits de la domotique,
- Très puissant,
- Possibilité de scripter des scénarios en utilisant un ensemble de commandes,
- Compatibilité native (depuis peu) de Google Assistant, ce qui va nous permettre de contrôler par la voix notre habitat,
- Large communauté.

---

[5] : <https://www.openhab.org>

### c. Critiques

Parmi les inconvénients de cette solution nous citons :

- L'outil n'est pas simple à appréhender

## 5. Analyse de la solution OpenRemote

### a. Description

C'est une solution domotique lancée en 2009, elle permet de relever les défis de l'intégration entre plusieurs protocoles et solutions disponibles pour la domotique et de proposer des outils de visualisation.



Figure 8 : Interface OpenRemote

### b. Avantages

Parmi les avantages de la solution OpenRemote nous citons :

- Plateforme open Source

### c. Critiques

Parmi les inconvénients de cette solution nous citons :

- Plateforme open Source mais n'est pas commercialisé.

## Synthèse

Nous avons étudié les plateformes les plus connus concernant le domaine de la domotique à savoir la sécurité et la flexibilité qui répondent à notre besoin et pour dégager surtout les problèmes rencontrés par les utilisateurs de ses plateformes afin de choisir avec laquelle on va travailler.



Le tableau dans l'annexe est un tableau comparatif des différentes plateformes afin de choisir l'approprié. (Voir annexe A)

### **III. Solution proposée**

Nombreuses solutions ne cessent d'apparaître jour après jour pour alléger les différents accidents domestiques.

En fait, l'idée la plus adoptée aujourd'hui pour minimiser ou pourquoi pas éliminer définitivement ces accidents est basée sur la disponibilité de matérielles.

D'ailleurs, en profitant des avancements en technologies de communication, **les solutions domotiques** deviennent capable de gérer des maisons ou encore des bâtiments d'une manière autonome et avec intervention humaine en cas de besoin.

Dans ce cadre intervient notre projet de fin d'études qui consiste à développer en premier lieu un système qui sert à contrôler une habitation et ceci à travers un middleware qui joue le rôle d'un intermédiaire entre les capteurs et les actionneurs et qui doit être installé dans un Gateway qui joue le rôle d'un serveur dans notre solution.

En second lieu, nous proposons de développer un système d'information pour permettre à l'habitant de consulter toutes les informations concernant sa maison ainsi qu'un système de contrôle pour permettre l'intervention humaine.

Sans oublier qu'une diffusion en temps réel de l'information entre ces trois systèmes doit être assurée.

### **IV. Méthodologies de travail**

Afin de choisir la méthodologie à adopter pour démarrer le développement de notre projet, nous nous sommes trouvées face à deux grandes familles de méthodes de gestion de projet : les méthodes classiques et les méthodes agiles.

Pour comprendre les finalités et les différences qu'elles présentent, tout d'abord, définissons-les :

D'une part, les méthodes Agiles [6] (dont les plus utilisées aujourd'hui sont les

---

[6] : <https://www.heflo.com/fr/blog/agile/methode-agile-scrum/>

méthodes Scrum, Kanban et Extrem Programming) procèdent par étape avec des objectifs à court terme.

Le client, garant du produit (Product Owner), est placé au centre des démarches. Le besoin métier est découpé en « User Stories » développées de manière itérative. Les projets sont découpés en « sprints », qui commencent par une vérification de la planification opérationnelle et se terminent par une démonstration de ce qui a été achevé, puis une rétrospective.

D'autre part, les méthodes classiques [7] ou encore les processus unifiés (2TUP) se découpent en plusieurs phases identifiées dès le démarrage du projet : étude préliminaire, définition des besoins fonctionnels et leurs analyse, définition des besoins techniques, conception générale puis détaillée, réalisation /développement, tests et mise en production, d'où la validation. Néanmoins, afin de faire le bon choix pour bien gérer notre projet et arriver à nos fins, il fallait bien comprendre dans quels cas chacune des deux méthodes est choisie.

Le mode agile est généralement choisi dans les gros projets qui présentent un niveau élevé d'incertitude.

Par contre, les méthodes classiques sont les plus adaptées si le projet est clair dès le début avec une idée précise et un cahier des charges et planning très bien détaillés.

## **1. Méthodologie choisie**

Notre plan de travail est approuvé et complété dès le début, en outre, l'entreprise d'accueil nous a également garanti que les besoins resteront inchangeables. En d'autres termes, toutes les étapes de notre projet sont prévisibles. C'est pourquoi la gestion en mode processus unifié répond parfaitement à nos besoins.

En guise de conclusion, notre choix va être certainement le processus unifié (2TUP) vu qu'elle est la plus pertinente et adéquate pour gérer notre projet de fin d'études.

---

[7] : <https://www.26academy.com/cours/les-methodes-claÖ-methodes-agiles/>

## **Conclusion**

Après avoir présenté le cadre de notre projet qui s'est articulé autour de 4 principaux axes : poser la problématique pour arriver au cadre général du projet, présenter et critiquer l'existant, proposer une solution, préciser la méthodologie à adopter.

Le chapitre qui suit s'intéresse à l'analyse et la spécification des besoins.

***Chapitre III :***  
***Analyse et spécification des***  
***besoins***

## **Introduction**

Ce chapitre est consacré à expliquer globalement ce que l'application est censée de faire. Tout d'abord, d'écrire l'application d'une manière générale. Ensuite, présenter l'ensemble des besoins fonctionnels et les besoins techniques du système. Après, le tout sera mieux clarifié et mis en valeur à travers un diagramme de cas d'utilisation général. Enfin, étudier les matériels qui répondent à nos besoins.

### **I. Description générale**

#### **1. Objectif de l'application**

L'objectif principale est de surmonté le problème des accidents domestiques qui s'augmente jour après jours.

#### **2. Fonctionnement de l'application**

Notre solution s'occupe de gérer toutes sorte d'accident et encore la confortabilité de l'utilisateur, tout en lui envoyer des alertes et des notifications pour lui informer ce que ce passe-t-il chez lui par le système d'information (Application Android) qui d'écrit en détaille l'état de sa maison et il peut intervenir par une interface web (Application Web) pour faire la gestion et le contrôle de la maison.

### **II. Etude des besoins**

#### **1. Analyse des besoins fonctionnels**

Lors de l'analyse des besoins, nous avons commencé par dégager les besoins fonctionnels de notre système et qu'est-ce que ce dernier devra faire exactement.

En effet, les besoins fonctionnels de notre application se divisent entre 3 systèmes :  
Un système permettant l'acquisition et la gestion en temps réel :

- La gestion d'ouverture et de fermeture des volets des fenêtres,
- La gestion des éclairages,
- La régulation de la température par la gestion d'aération,
- La gestion de sécurité contre toute sorte d'intrusion,
- L'acquisition de la température et de l'humidité.

Un système d'information :

- Alerter/Notifier l'utilisateur de ce que se passe à l'intérieure de la maison,
- Informer en temps réel l'utilisateur par l'état de sa maison,

Un système automatique :

- Exécute les scénarios automatiquement sans besoins d'une intervention humaine.

## **2. Analyse des besoins techniques**

Après avoir cité les besoins fonctionnels attendues, nous avons passé directement à la phase concernant l'analyse des besoins techniques qui sont également les besoins non fonctionnels de notre système.

Les besoins techniques de notre système se décrivent comme suit :

- **La sécurité** : le système doit être sécurisé et protégé par un boitier et l'application doit être sécurisée car les informations ne doivent pas être accessibles à tout le monde.
- **La fiabilité** : le système doit être en temps réel.
- **Ergonomie et souplesse** : Notre application doit offrir une interface claire, et ergonomique exploitable par l'utilisateur
- **L'autonomie** : le système doit être autonome et interagit sans besoin d'intervention humaine.
- **La sauvegarde** : toutes les données doivent être sauvegarder dans la base de données.

## **III. Diagramme de cas d'utilisation**

Après avoir présenté les besoins fonctionnels et les besoins non fonctionnels de notre application, nous allons passer à dégager les différents cas d'utilisations tout en mettant en relief les interactions fonctionnelles entre les acteurs et le système.

### **1. Identification des acteurs**

Cette partie s'intéresse à présenter les différents acteurs qui peuvent interagir avec le système. Un acteur est l'idéalisation d'un rôle joué par une personne externe, un processus, un dispositif matériel ou même un autre système qui interagit directement avec le système étudié.

Dans notre cas, l'habitant qui a le droit de manipuler toutes les fonctionnalités proposées par l'application.

## 2. Diagramme de cas d'utilisation général

Dans cette partie, nous présentons le diagramme de cas d'utilisation général du système :

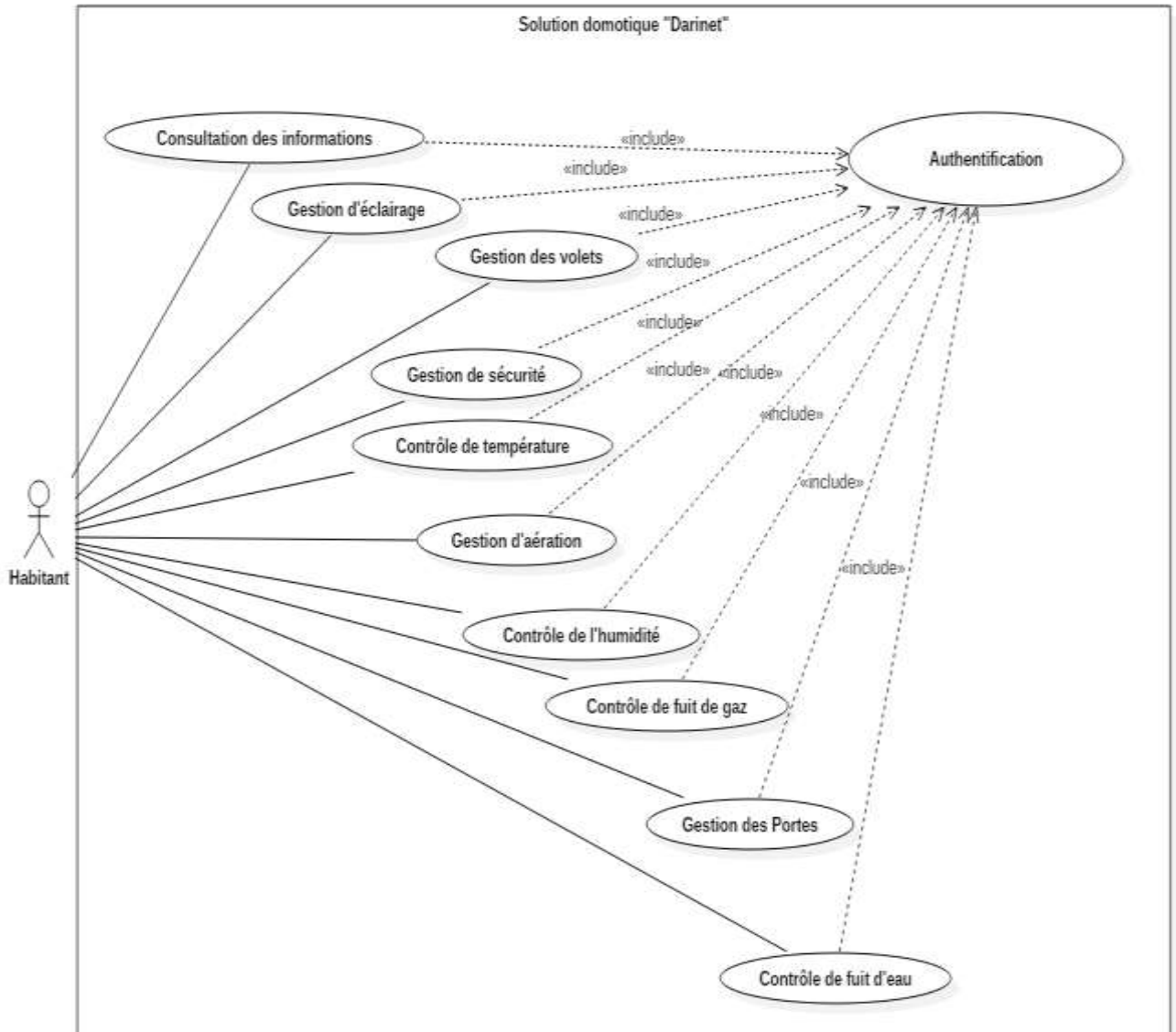


Figure 9 : Diagramme des cas d'utilisation générale du système

## IV. Etude matérielle

Après l'étude et l'analyse des besoins fonctionnelles et techniques, nous avons remarqué que le côté matériel de notre application se résume en nœud de capteurs/actionneurs et d'un Gateway sous forme d'une carte électronique.

## 1. Etude des capteurs

Les capteurs qui peuvent exister dans la domestique sont très nombreux, mais dans notre cas, les capteurs qui peuvent satisfaire nos besoins sont :

- Capteur température et humidité :

Le tableau ci-dessous contient les différents capteurs de température et d'humidité :

**Tableau 1 : Tableau comparatif des capteurs de température & humidité**

				
	Dht21	LM358	Dht11	Dht22
<b>Type</b>	Température & humidité	Température	Température & humidité	Température & humidité
<b>Alimentation</b>	3.3 - 5.2V	4 – 30V	<b>3.3-5.5V</b>	3,3 à 6 V
<b>Consommation</b>	Faible	Faible	0.5 mA 2.5 mA max	1,5 mA Au repos : 50 µA
<b>Température mesurable</b>	De -40 à 80°C ±0.5°C	De -55° à +150°C	0°C à 50°C ± 2°C	-40 à +80 °C
<b>Humidité mesurable</b>	De 0 à 99.9% 3% RH	-	20-90% RH ±5% RH	0 à 100 % RH
<b>Cout</b>	14.59 €	2.33 €	4.08€	7.29€

Après une étude sur ces différents capteurs, nous avons distingué que l'intervalle mesurable de température et d'humidité varie du capteur à un autre. Mais suivant notre besoin, nous avons choisi de travailler avec le capteur dht11 puisque son intervalle de détection satisfait nos besoin, son prix est inférieur au prix de dht22 et qu'il capte à la fois la température ainsi que l'humidité.






- Capteur détecteur de mouvement :

[8]<https://www.electroschematics.com/6393/lm35-datasheet/>



Le tableau ci-dessous contient quelques détecteurs de mouvement [9] :

**Tableau 2 : Tableau comparatif des capteurs détecteurs de mouvement**

	 Détecteur De Mouvement PIR180	 Détecteur de mouvement encastré	 Steinel détecteur de mouvement	 Détecteur De Mouvement PIR	 Capteur PIR
<b>Alimentation</b>	230 Volts	230V	230 Volts	230v	5V-20V
<b>Consommation</b>	1000 Watts	0,5W	500 Watts	1200 watts	65mA
<b>Portée</b>	12 m	2 à 9 m	10 m	-	7m
<b>Angle de détection</b>	180°	160°	120°	180°	110°
<b>Cout</b>	11,99 €	14,99 €	24,98 €	9,14 €	1,02 €
<b>Poids</b>	159 grammes	81,6 g	141 g	200 grammes	6g
<b>Référence</b>	<a href="https://www.detecteur-mouvement.com/">https://www.detecteur-mouvement.com/</a>				

Après l'étude sur les différents capteurs détecteurs de mouvement existantes nous avons interprété que tous ces détecteur travail à base du capteur PIR ce qui implique que ce capteur qui répond à notre besoin.








- **Capteur de gaz**

Le tableau ci-dessous contient quelques capteurs de gaz :

---

[9 ] <https://www.detecteur-mouvement.com/>

Tableau 3 : Tableau comparatif des capteurs de gaz

								
Type de gaz	-Méthane, -Butane, -la fumée	-Alcool -Ethanol -la fumée	-Méthane - gaz CNG	-Gaz naturel -LPG	-LPG, -Butane	-Monoxyde de carbone	-Hydrogène	-Qualité d'air
Alimentation	5v	5v	5V	5V	5V	5V	5V	5V
Consommation	0.16A	0.1A	0.15A	0.1A	0.1A	0.1A	0.1 A	0.15A
Puissance	800mW	>750 mW	900 mW	>800mW	>750mw	350mW	800mW	<800 mW
Prix	15TND	14 TND	14 TND	15 TND	20TND	15 TND	20 TND	28TND
Références	<a href="http://image.dfrobot.com/image/data/SEN0127/MQ-2.pdf">http://image.dfrobot.com/image/data/SEN0127/MQ-2.pdf</a>	<a href="http://fr.hobbytronics.co.uk/mq3-alcohol-gas-sensor/">http://fr.hobbytronics.co.uk/mq3-alcohol-gas-sensor/</a>	<a href="https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-4.pdf">https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-4.pdf</a>	<a href="https://www.gotronic.fr/ark-t-capteur-de-gaz-mq5-grove-101020056-22960.htm">https://www.gotronic.fr/ark-t-capteur-de-gaz-mq5-grove-101020056-22960.htm</a>	<a href="https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-6.pdf">https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-6.pdf</a>	<a href="https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf">https://www.sparkfun.com/datasheets/Sensors/Biometric/MQ-7.pdf</a>	<a href="https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Biometric/MQ-8.pdf">https://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Biometric/MQ-8.pdf</a>	<a href="https://www.robotshop.com/eu/fr/capteur-gaz-mq-135.html">https://www.robotshop.com/eu/fr/capteur-gaz-mq-135.html</a>

Après l'étude sur les différents capteurs de gaz existantes nous avons interprété que chaque capteur détecte différent type de gaz. Du coup, nous avons trouvé que le choix de capteur MQ135 est le choix le plus optimal puisqu'il sert à détecter une large gamme de gaz, il est sensible aux principaux polluants présents dans l'atmosphère de la maison et il est le moins chers en le comparant avec les autres capteurs.

- Capteur détecteur niveau d'eau :

Le tableau ci-dessous contient quelques détecteurs de niveau d'eau <sup>[10]</sup> :

<sup>[10]</sup> : <https://www.gotronic.fr/cat-niveau-1136.htm>






					
	Détecteur de niveau NO FLSW1	Détecteur de niveau vertical NF MLS7249	Capteur de niveau d'eau ST045	Détecteur de niveau d'eau Grove	Capteur de niveau Gravity SEN0204
<b>Alimentation</b>	100 Vcc max	24 Vcc	5 Vcc	5 Vcc	5 à 24 Vcc
<b>Consommation</b>	0,5 A maxi	0,5 A maxi	>20 mA	Faible consommation	5 mA
<b>Température de travail</b>	De 0 à 70 °C	>20 °C à 100 °C	De 10°C à 30°C	De 0°C à +40°C	De 0 à 105 °C
<b>Coût</b>	7,50 €	10,90 €	2,40 €	3,05 €	11,45 €

Tableau 4 : Tableau comparatif des capteurs détecteurs de niveau d'eau




## 2. Etude des actionneurs

Cette partie présente les divers actionneurs existants répondant à nos besoins :

- Les moteurs : pas à pas, courant continu, servomoteurs ...

Le tableau ci-dessous contient quelques moteurs :

Tableau 5 : Tableau comparatif des différents moteurs

			
	Servomoteurs	Moteur pas à pas	Moteur à courant continu
<b>Alimentation</b>	5V	5V	3-6V
<b>Consommation</b>	550 mA	25 mA	70mA
<b>Vitesse</b>	0.1s / 60 °	-	1:48s
<b>Couples</b>	2.5kg / cm	300 gf.cm	-
<b>Rotation</b>	0 ° - 180°	1,8 ° 0,9 ° demi-marche	360°
<b>Poids</b>	9g	0,24 kg	
<b>Cout</b>	2,32 €	5,70 €	
<b>Références</b>	<a href="https://components101.com/servo-motor-basics-pinout-datasheet">https://components101.com/servo-motor-basics-pinout-datasheet</a>	<a href="http://robocraft.ru/files/datasheet/28BYJ-48.pdf">http://robocraft.ru/files/datasheet/28BYJ-48.pdf</a>	

Après une étude sur les actionneurs, nous avons traité la phase des moteurs, nous avons choisi de travailler avec les servomoteurs pour les portes ainsi que les volets.

- Buzzer
- Ventilateur

### 3. Etude des cartes

Dans notre cas, nous avons besoin de deux cartes électroniques, une pour former le nœud de capteurs/actionneurs, et l'autre carte joue le rôle d'un Gateway.

Parmi des dizaines de cartes disponibles sur le marché, les familles des cartes les plus dominants qui peuvent répondre à nos besoins sont :

Tableau 6 : Tableau d'étude sur les cartes électroniques

	Raspberry pi [11]	Arduino UNO [12]	Arduino Mega [13]	Arduino nano [14]	Beagle Board black [15]
<b>Tension d'alimentation</b>	3.3V-5V	5V	5V	5V	5V
<b>Interface de communication</b>	UART, SPI, TWI	SPI, TWI	SPI, I2C	SPI, I2C	UART, SPI, I2C
<b>Courant de fonctionnement</b>	54mA	50 mA	50 mA	19 mA	210-460mA
<b>Fréquence</b>	1.2GHz	16 MHz	16 MHz	16 MHz	1GHz
<b>Nombre de pins</b>	40 pins d'E/S	14 pins numériques 6 pins analogique	54 pins numériques 16 pins analogiques	14 pins numériques 8 pins analogiques	58 pins numériques 7 pins analogiques
<b>PWM</b>	GPIO12, GPIO13, GPIO18, GPIO19	pin3, 5, 6, 9, 10, et 11	Du pin0 jusqu'au pin13	pin3, 5, 6, 9, 10, et 11	pin13, 14, 16, 19, 21, 22, 28, 29, 31, 34, 36, 42, 45, pin46
<b>Microprocesseur</b>	Broadcom BCM2837 64bit Quad Core Processor	ATmega328	ATmega2560	Atmel ATmega168 or ATmega328	Sitara AM3359AZCZ100
<b>Mémoire flash</b>	16Gbytes SSD memory card	32 KB (ATmega328)	256 KB	16 Ko (ATmega168) ou 32 Ko (ATmega328)	-
<b>Mémoire SRAM</b>	-	2 kB	8 kB	2 ko	-
<b>Mémoire EEPROM</b>	-	1 kB	4 kB	1 ko	-
<b>Courant par pin</b>	16mA	40 mA	40 mA	40 A	-

[11] : <https://components101.com/microcontrollers/raspberry-pi-3-pinout-features-datasheet>

[12] : <https://www.farnell.com/datasheets/1682209.pdf>

[13] : <http://www.mantech.co.za/datasheets/products/A000047.pdf>

[14] : <http://www.farnell.com/datasheets/1682238.pdf>

[15] : <https://components101.com/microcontrollers/beaglebone-black-pinout-datasheet> <http://beagleboard.org/Support/bone101>

## **Conclusion**

Dans ce chapitre nous avons analysé les besoins fonctionnels et les besoins techniques attendues par notre solution. Nous avons commencé par les capturer, les définir et les expliquer, puis, nous avons présenté le diagramme de cas d'utilisation générale. Puis, nous avons fini par l'étude matérielle afin de choisir le matériel nécessaire pour la réalisation de la solution.

# *Chapitre IV:* *Conception*

## Introduction

Nous aborderons dans ce chapitre notre solution conceptuelle. D'abord, nous commençons par présenter l'architecture générale de notre solution pour passer ensuite à la conception détaillée des besoins.

### I. Architecture globale de la solution

Dans cette partie, nous présentons l'architecture globale de notre solution.

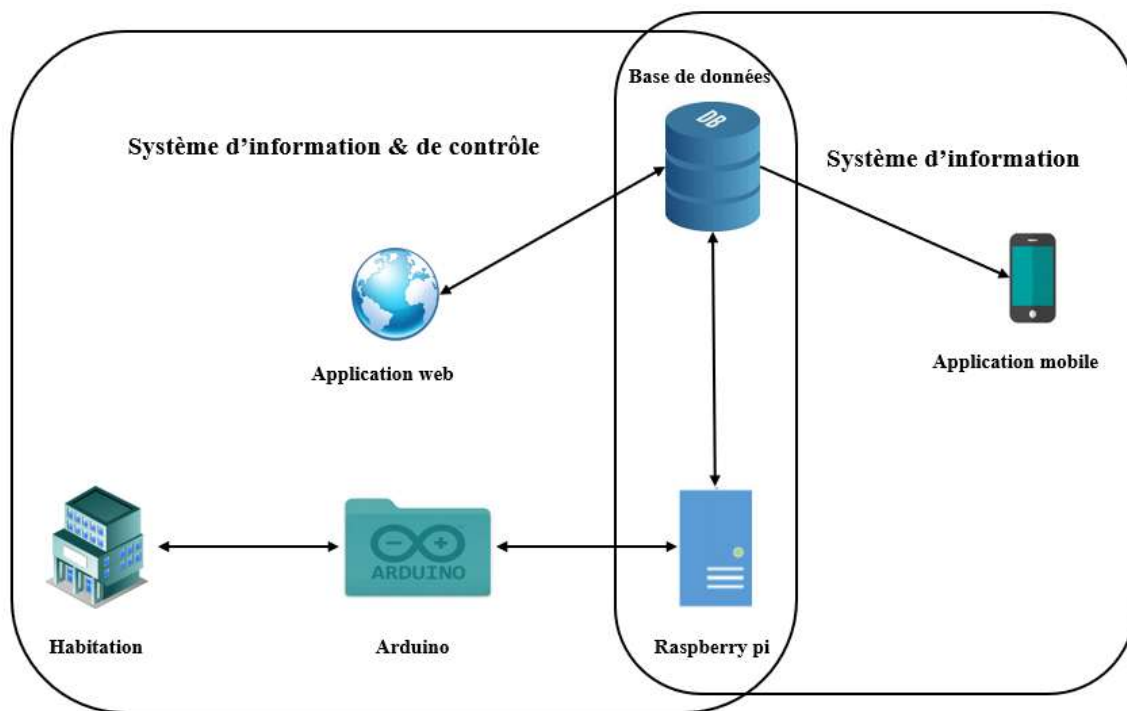


Figure 10 : Architecture globale de la solution

Tout d'abord, notre solution est décomposée en 3 systèmes :

- Un système d'information : Grâce à une application mobile sur son smartphone, nous donnons la chance à l'habitant de consulter toutes les informations concernant l'état de sa maison.
- Un système d'information et de contrôle : Avec une application web, l'habitant peut intervenir et contrôler sa maison.
- Un système autonome : Sans besoin d'intervention humaine, notre système peut gérer la maison de l'habitant et lui envoi des alertes ou des notifications pour lui informer de l'état de sa maison pour chaque action.



## 1. Architecture matérielle

La figure ci-dessous présente l'architecture logicielle de la solution :

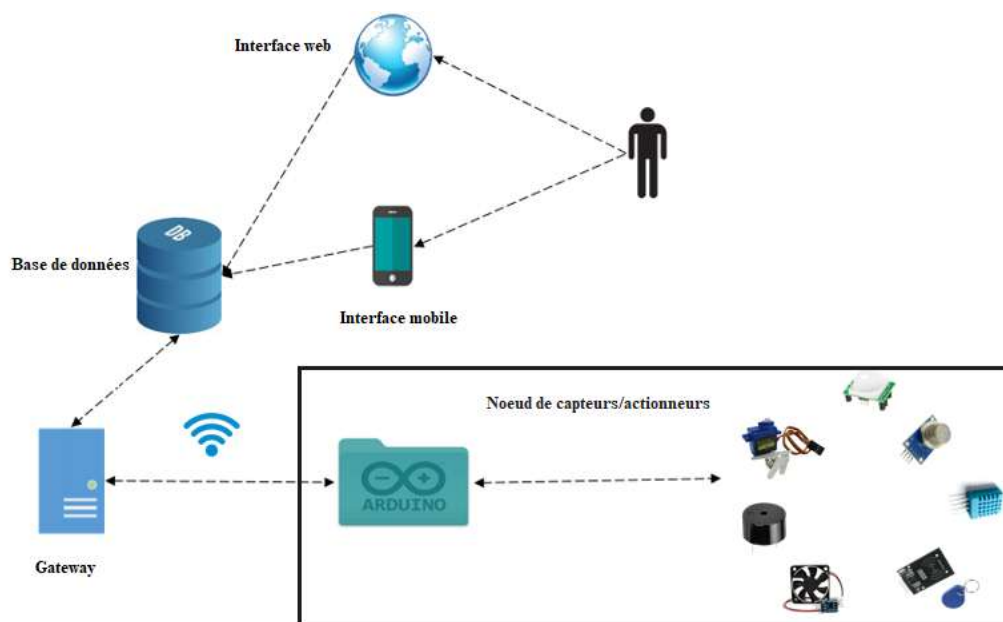


Figure 11 : Schéma d'architecture matérielle

## 2. Architecture logicielle

La figure ci-dessous présente l'architecture logicielle de la solution :

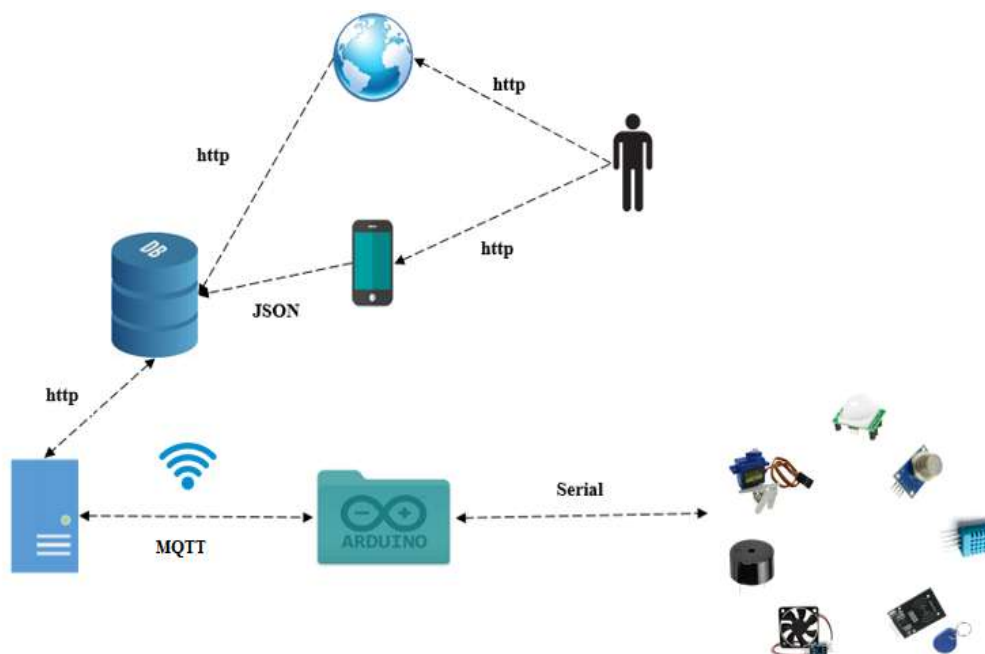


Figure 12 : Schéma d'architecture logicielle

Notant que :

- **JSON (JavaScript Object Notation)** : est un format léger d'échange de données, il permet d'afficher les données sous forme XML, utilisé pour intégrer et automatiser les applications en ligne.
- **MQTT (Message Queuing Telemetry Transport)** : est un protocole de messagerie publish-subscribe basé sur le protocole TCP/IP.
- **Série** : permet de communiquer les capteurs/actionneurs avec le microcontrôleur.
- **Http (Hypertext Transfer Protocol)** : Protocole de transmission permettant à l'utilisateur d'accéder à des pages web par l'intermédiaire d'un navigateur.

Expliquant mieux l'architecture logicielle de notre solution :

La maison de l'habitant est équipée par des divers capteurs et des divers actionneurs qui sont tous liés en série avec un microcontrôleur pour assurer l'échange des informations et des données ce bloc on l'appelle « nœud de capteurs/actionneurs ».

Ce nœud est responsable de collecter les données captées par les divers capteurs (capteur détecteur de mouvement, capteur température et humidité, capteur gaz, capteur niveau d'eau...) et les envoyer au Gateway « Raspberry pi » par wifi via « MQTT » le protocole de messagerie publish-subscribe TCP/IP. Ces données reçues s'affichent sur le Gateway grâce à un middleware « OpenHab » installé dedans et qui sauvegarde toutes l'information reçue et captée dans une base de données.

Notre solution peut être accessible avec deux interfaces, ce qui implique que l'habitant peut accéder pour contrôler sa maison avec une interface web ou bien avec application Android sur son smartphone grâce au protocole de transmission « http ». Ces deux interfaces interagissent avec la base de données qui sauvegarde toutes sortes d'informations reçues du Gateway par la technologie « JSON » qui est responsable d'intégrer et automatiser les applications en ligne.

## II. Vue statique

### 1. Diagramme de cas d'utilisation

#### 1.1. Diagramme de cas d'utilisation : Gestion des éclairages

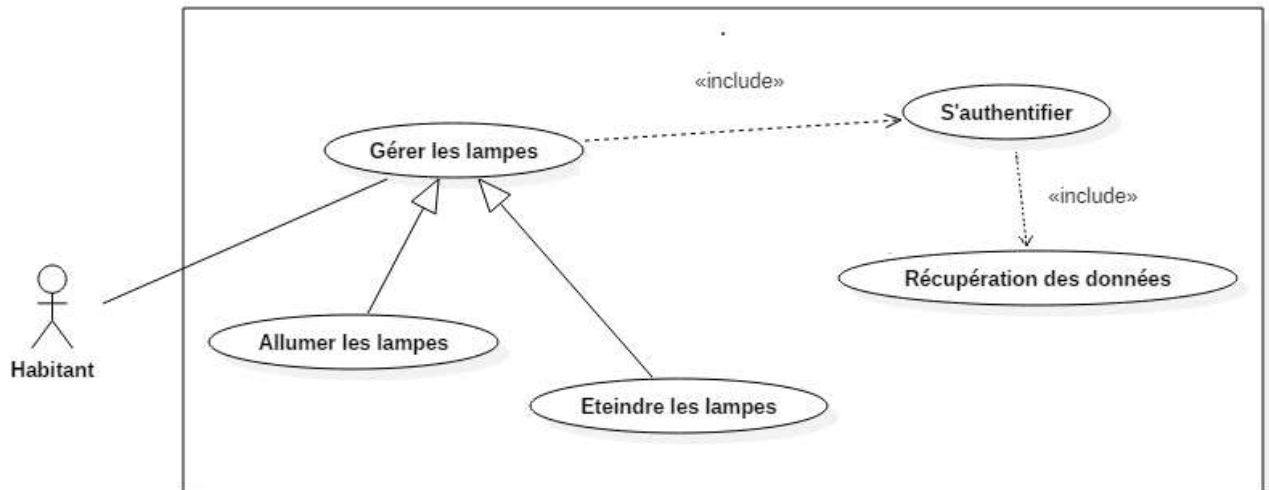


Figure 13 : Diagramme cas d'utilisation « Gestion d'éclairage »

#### 1.1.1. Description textuelle

##### a. Allumer les lampes

Tableau 7 : Description textuelle de cas d'utilisation « Allumage des lampes »

<b>Cas d'utilisation</b>	Allumer les lampes
<b>Objectif</b>	Allumer les lampes du maison
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Lampe éteinte,</li> <li>- L'état de la lampe dans la base de données est « OFF ».</li> </ul>
<b>Postcondition</b>	Lampe allumée
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1. Choisir la lampe à allumer,</li> <li>2. Cliquer sur le bouton switcher approprié pour assurer l'allumage de la lampe.</li> </ol>
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet

## b. Eteindre les lampes

Tableau 8 : Description textuelle de cas d'utilisation « Extinction des lampes »

<b>Cas d'utilisation</b>	Eteindre les lampes
<b>Objectif</b>	Eteindre les lampes du maison
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Lampe allumée,</li> <li>- L'état de la lampe dans la base de données est « ON ».</li> </ul>
<b>Postcondition</b>	Lampe éteinte
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1. Choisir la lampe à éteindre,</li> <li>2. Cliquer sur le bouton switcher approprié pour assurer l'extinction de la lampe.</li> </ol>
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet

## 1.2. Diagramme de cas d'utilisation : Gestion de sécurité

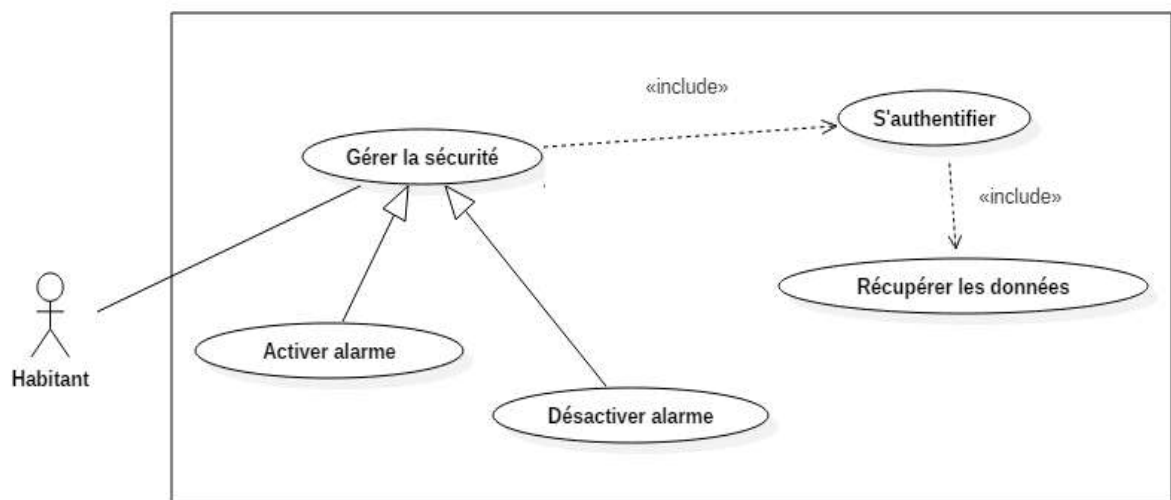


Figure 14: Diagramme de cas d'utilisation « Gestion de sécurité »

### 1.2.1. Description textuelle

#### a. Activer alarme

Tableau 9 : Description textuelle de cas d'utilisation « Activer alarme »

<b>Cas d'utilisation</b>	Activer alarme
<b>Objectif</b>	Activer l'alarme
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Alarme activée,</li> <li>- L'état de l'alarme dans la base de données est « OFF ».</li> </ul>
<b>Postcondition</b>	Alarme activée
<b>Scénario nominal</b>	1. Cliquer sur le bouton switcher dans l'application web pour activer l'alarme.
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet

#### b. Désactiver alarme

Tableau 10 : Description textuelle de cas d'utilisation « Désactiver alarme »

<b>Cas d'utilisation</b>	Désactiver alarme
<b>Objectif</b>	Désactiver l'alarme
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Alarme désactivée,</li> <li>- L'état de l'alarme dans la base de données est « ON ».</li> </ul>
<b>Postcondition</b>	Alarme désactivée
<b>Description</b>	1. Cliquer sur le bouton switcher dans l'application web pour désactiver l'alarme.
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Aucun

### 1.3. Diagramme de cas d'utilisation : Gestion des portes

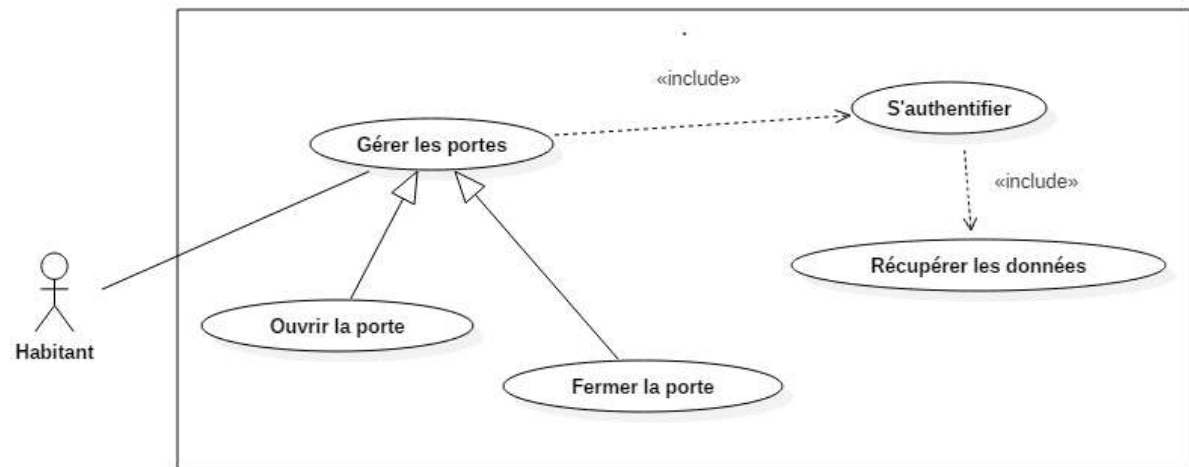


Figure 15 : Diagramme de cas d'utilisation « Gestion des portes »

#### 1.3.1. Description textuelle

##### a. Ouvrir la porte

Tableau 11 : Description textuelle de cas d'utilisation « Ouvrir la porte »

<b>Cas d'utilisation</b>	Ouvrir la porte
<b>Objectif</b>	Ouvrir la porte d'une chambre choisie
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Porte ouverte,</li> <li>- L'état de la porte dans la base de données est « OFF ».</li> </ul>
<b>Postcondition</b>	Porte ouverte
<b>Scénario nominal</b>	1. Cliquer sur le bouton switcher dans l'application web pour ouvrir la porte.
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet

## b. Fermer la porte

Tableau 12 : Description textuelle de cas d'utilisation « Fermer la porte »

<b>Cas d'utilisation</b>	Fermer la porte
<b>Objectif</b>	Fermer la porte d'une chambre choisie
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Porte fermée,</li> <li>- L'état de la porte dans la base de données est « ON ».</li> </ul>
<b>Postcondition</b>	Porte fermée
<b>Scénario nominal</b>	1. Cliquer sur le bouton switcher dans l'application web pour fermer la porte.
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet

## 1.4. Diagramme de cas d'utilisation : Gestion des volets

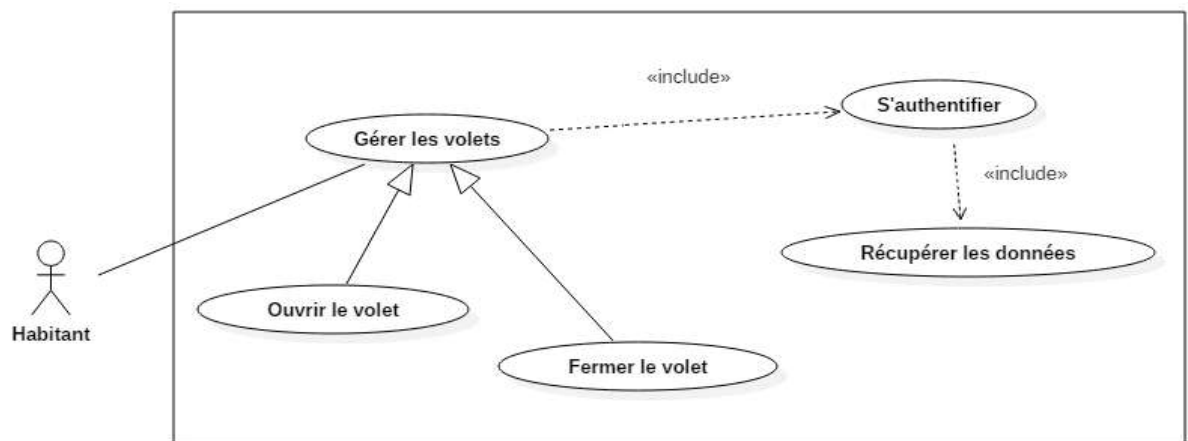


Figure 16 : Diagramme de cas d'utilisation « Gestion des volets »

### 1.4.1. Description textuelle

#### a. Ouvrir le volet

Tableau 13 : Description textuelle de cas d'utilisation « Ouvrir le volet »

<b>Cas d'utilisation</b>	Ouvrir le volet
<b>Objectif</b>	Ouvrir le volet d'une chambre
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier</li> <li>- Volet ouvert,</li> <li>- L'état du volet dans la base de données est « OFF ».</li> </ul>
<b>Postcondition</b>	Volet ouvert
<b>Scénario nominal</b>	1. Cliquer sur le bouton switcher dans l'application web pour ouvrir le volet.
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet

#### b. Fermer le volet

Tableau 14 : Description textuelle de cas d'utilisation « Fermer le volet »

<b>Cas d'utilisation</b>	Fermer le volet
<b>Objectif</b>	Fermer le volet d'une chambre.
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Volet fermé,</li> <li>- L'état du volet dans la base de données est « ON ».</li> </ul>
<b>Postcondition</b>	Volet fermé
<b>Scénario nominal</b>	1. Cliquer sur le bouton switcher dans l'application web pour fermer le volet.
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet



## 1.5. Diagramme de cas d'utilisation : Gestion d'aération

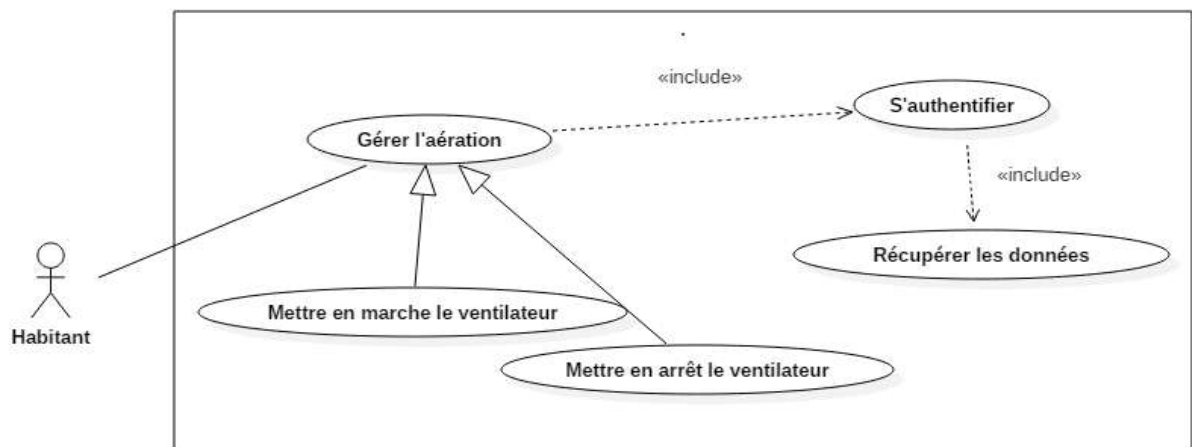


Figure 17 : Diagramme de cas d'utilisation « Gestion d'aération »

### 1.5.1. Description textuelle

#### a. Mettre en marche le ventilateur

Tableau 15 : Description textuelle de cas d'utilisation « Mettre en marche le ventilateur »

<b>Cas d'utilisation</b>	Mettre en marche le ventilateur
<b>Objectif</b>	Démarrer le système d'aération.
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Ventilateur en marche,</li> <li>- L'état du ventilateur dans la base de données est « OFF ».</li> </ul>
<b>Postcondition</b>	Ventilateur en marche
<b>Scénario nominal</b>	1. Cliquer sur le bouton switcher dans l'application web pour mettre en marche le ventilateur.
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet.

## b. Mettre en arrêt le ventilateur

Tableau 16 : Description textuelle de cas d'utilisation « Mettre en arrêt le ventilateur »

<b>Cas d'utilisation</b>	Mettre en arrêt le ventilateur
<b>Objectif</b>	Arrêter le système d'aération.
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"> <li>- S'authentifier,</li> <li>- Ventilateur arrêté,</li> <li>- L'état du ventilateur dans la base de données est « ON ».</li> </ul>
<b>Postcondition</b>	Ventilateur arrêté
<b>Scénario nominal</b>	1. Cliquer sur le bouton switcher dans l'application web pour arrêter le ventilateur.
<b>Scénario alternatif</b>	Aucun
<b>Scénario d'erreur</b>	Vérifier la connexion internet.

## 1.6. Diagramme de cas d'utilisation : Consultation des informations

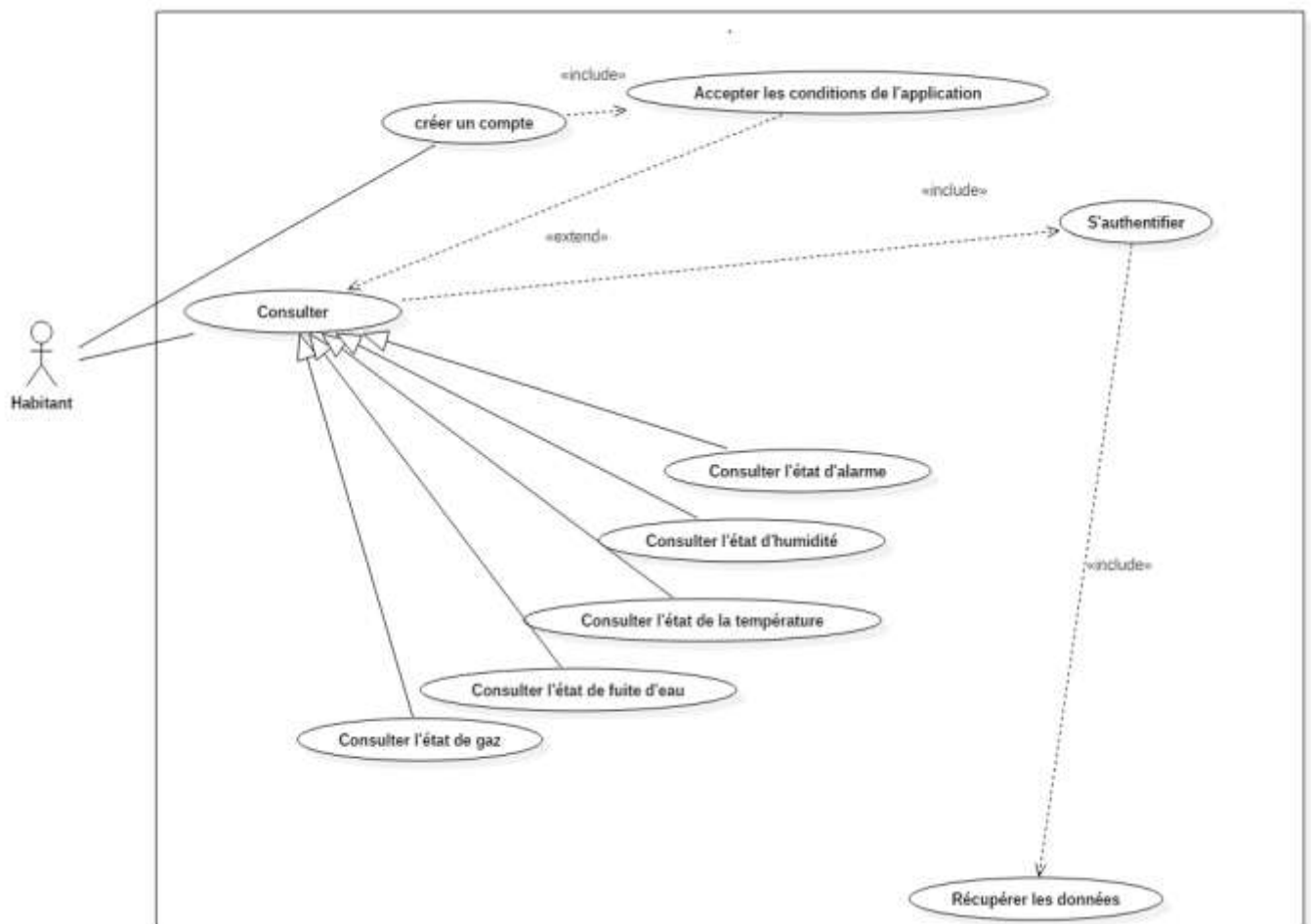


Figure 18 : Diagramme de cas d'utilisation « Consultation des informations »

### 1.6.1. Description textuelle

#### a. Créer un compte

Tableau 17 : Description textuelle de cas d'utilisation « Création du compte »

<b>Cas d'utilisation</b>	Créer un compte
<b>Objectif</b>	Créer un compte pour accéder à l'application
<b>Acteur</b>	Habitant
<b>Précondition</b>	- Aucun utilisateur est enregistré dans la base de données,
<b>Postcondition</b>	- Un utilisateur s'est ajouté à la base de données,
<b>Scénario nominal</b>	<ol style="list-style-type: none"><li>1. Démarrer l'application mobile,</li><li>2. Accepter les conditions de l'application,</li><li>3. Remplir les champs afin de pouvoir créer un compte,</li><li>4. Cliquer sur le bouton « Register » pour confirmer votre compte.</li></ol>
<b>Scénario alternatif</b>	<p>1'. Internet non activé :</p> <ul style="list-style-type: none"><li>- L'application affiche un message d'erreur</li><li>- L'utilisateur active l'internet</li><li>- Le cas d'utilisation reprend l'action 3 du scénario nominal.</li></ul>
<b>Scénario d'erreur</b>	<p>4'. Des champs manquants dans le formulaire de saisie lors d'inscription :</p> <ul style="list-style-type: none"><li>- Le cas d'utilisation reprend l'action 3 du scénario nominal.</li></ul>

**b. Accepter les conditions de l'application**

Tableau 18 : Description textuelle de cas d'utilisation « Accepter les conditions de l'application »

<b>Cas d'utilisation</b>	Accepter les conditions de l'application
<b>Objectif</b>	Accepter les conditions de l'application.
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"><li>- Lancer l'application mobile</li><li>- Connexion internet</li></ul>
<b>Postcondition</b>	Créer un compte
<b>Scénario nominal</b>	<ol style="list-style-type: none"><li>1. Démarrer l'application mobile,</li><li>2. Une interface contenant les conditions de notre application s'affiche,</li><li>3. Cliquer sur le bouton « Accepter tous »,</li></ol>
<b>Scénario alternatif</b>	<ol style="list-style-type: none"><li>3'. Cocher les conditions que vous pouvez l'accepter,</li><li>4'. Cliquer sur le bouton « Suivant »</li></ol>
<b>Scénario d'erreur</b>	<p>3 ''. L'utilisateur clique sur suivant sans cocher aucune condition :</p> <ul style="list-style-type: none"><li>- L'application affiche « veuillez cocher au moins une condition ».</li></ul>

**c. S'authentifier**

**Tableau 19 : Description textuelle de cas d'utilisation « Authentification »**

<b>Cas d'utilisation</b>	S'authentifier
<b>Objectif</b>	S'authentifier
<b>Acteur</b>	Habitant
<b>Précondition</b>	<ul style="list-style-type: none"><li>- Lancer l'application mobile</li><li>- Accepter les conditions de l'application,</li><li>- Compte utilisateur existant déjà.</li></ul>
<b>Postcondition</b>	Accéder à l'interface de consultation.
<b>Scénario nominal</b>	<ol style="list-style-type: none"><li>1. Tapez votre email,</li><li>2. Tapez votre mot de passe,</li><li>3. Cliquer sur le bouton « login ».</li></ol>
<b>Scénario alternatif</b>	<p>3'. Champ email vide :</p> <ul style="list-style-type: none"><li>- Le cas d'utilisation reprend l'action 1 du scénario nominal.</li></ul> <p>3'. Champ mot de passe vide :</p> <ul style="list-style-type: none"><li>- Le cas d'utilisation reprend l'action 1 du scénario nominal.</li></ul>
<b>Scénario d'erreur</b>	3''. L'application affiche : « Compte n'existe pas ».

## 2. Diagramme de déploiement

Le diagramme de déploiement est une vue statique qui sert à représenter l'utilisation de l'infrastructure physique par le système et la manière dont les composants du système sont répartis ainsi que les relations entre eux.

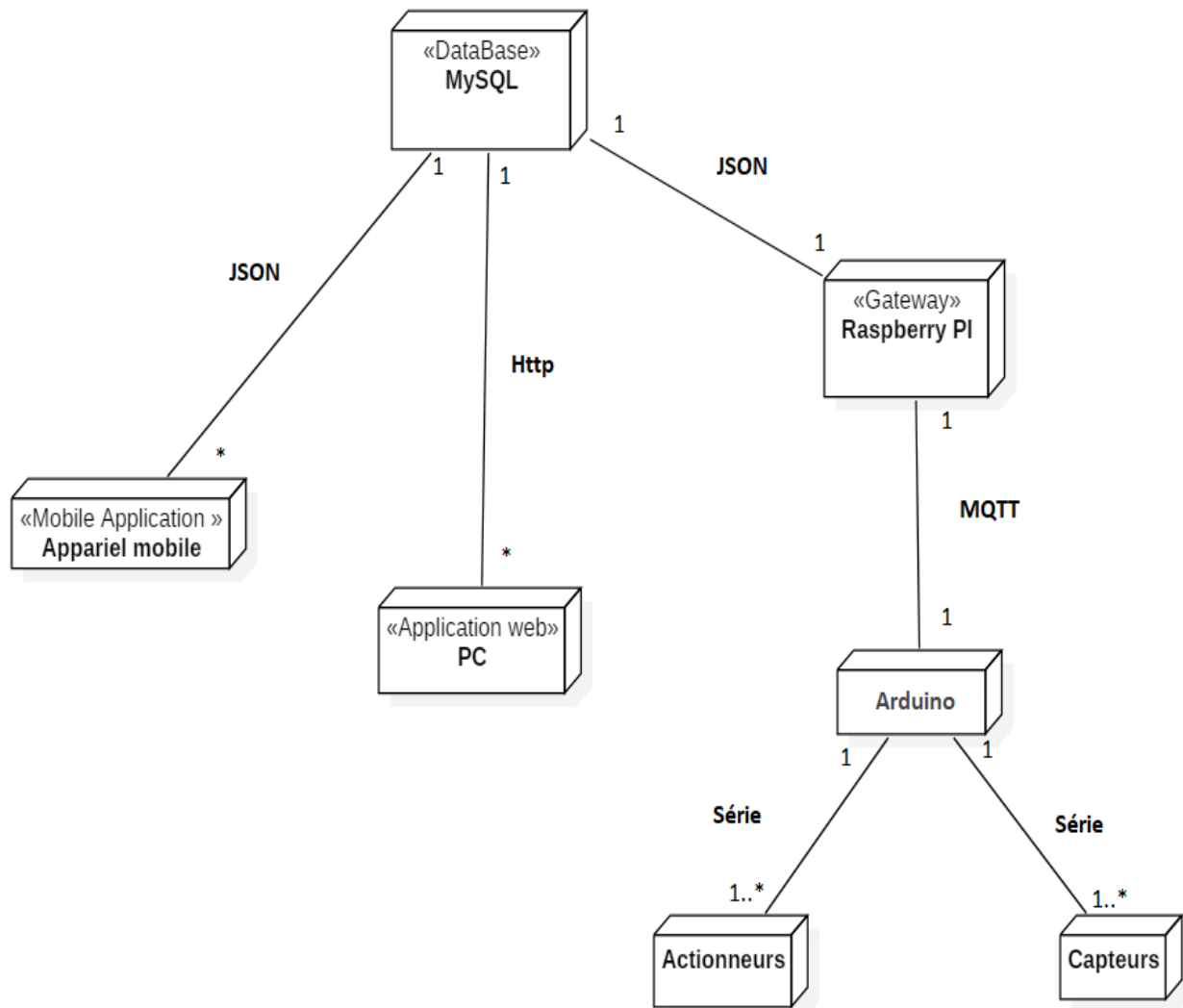
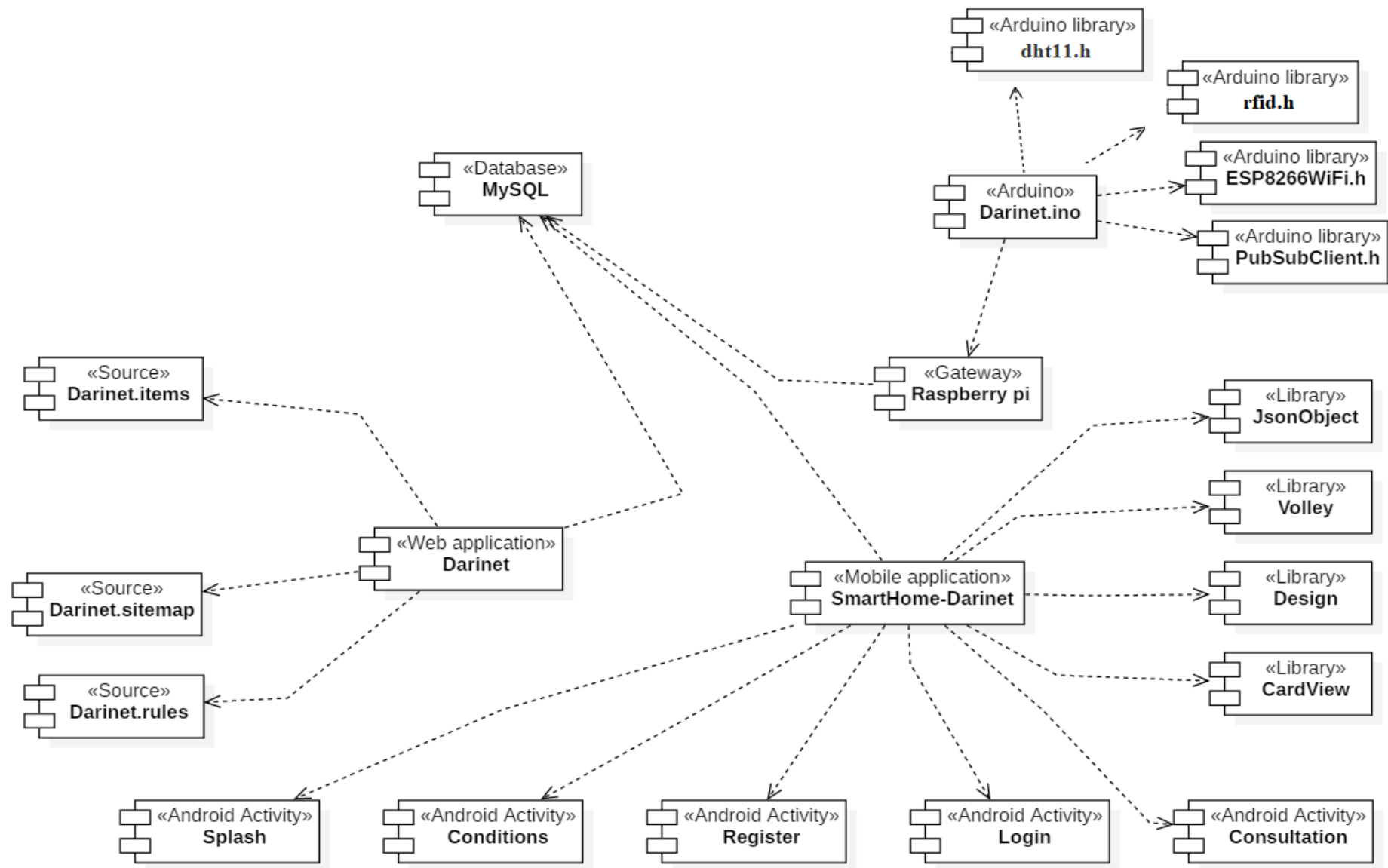


Figure 19 : Diagramme de déploiement

### 3. Diagramme de composants

Le diagramme de composants décrit l'organisation du système de point de vue éléments logiciels. Il permet de mettre en évidence les dépendances entre les composants.



### **III. Vue dynamique**

#### **1. Diagramme de séquence**

Un diagramme de séquence montre la séquence verticale des messages passés entre objets au sein d'une interaction.

##### **1.1. Application Mobile**

###### **a. Diagramme de séquence relatif au « Consultation des informations »**

La figure ci-dessous présente le diagramme de séquence de l'application Mobile relatif au cas d'utilisation « Consultation des informations » :



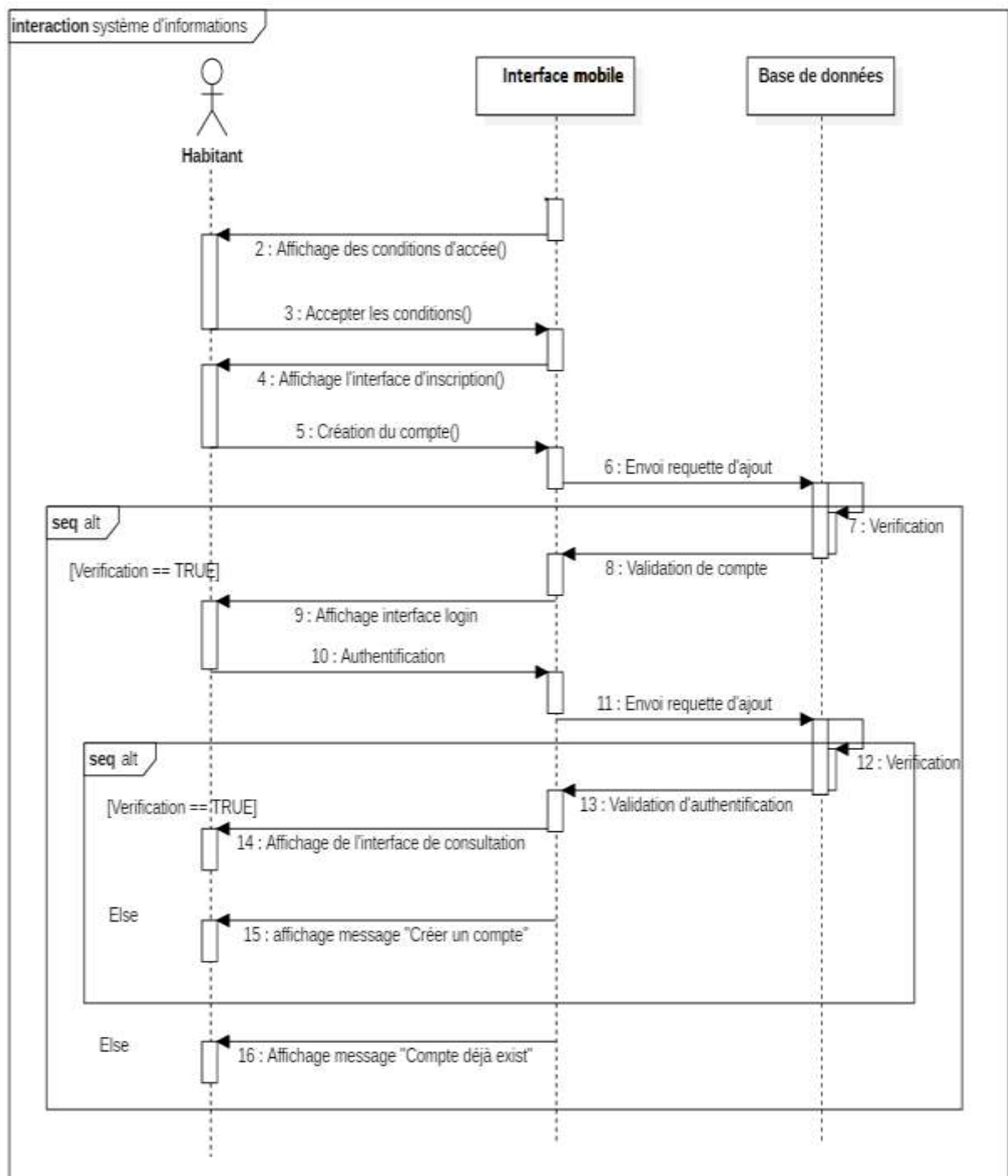


Figure 21 : Diagramme de séquence de l'application Mobile : cas d'utilisation « Consultation des informations »

## 1.2. Application Web

La figure ci-dessous présente le diagramme de séquence des cas d'utilisation :

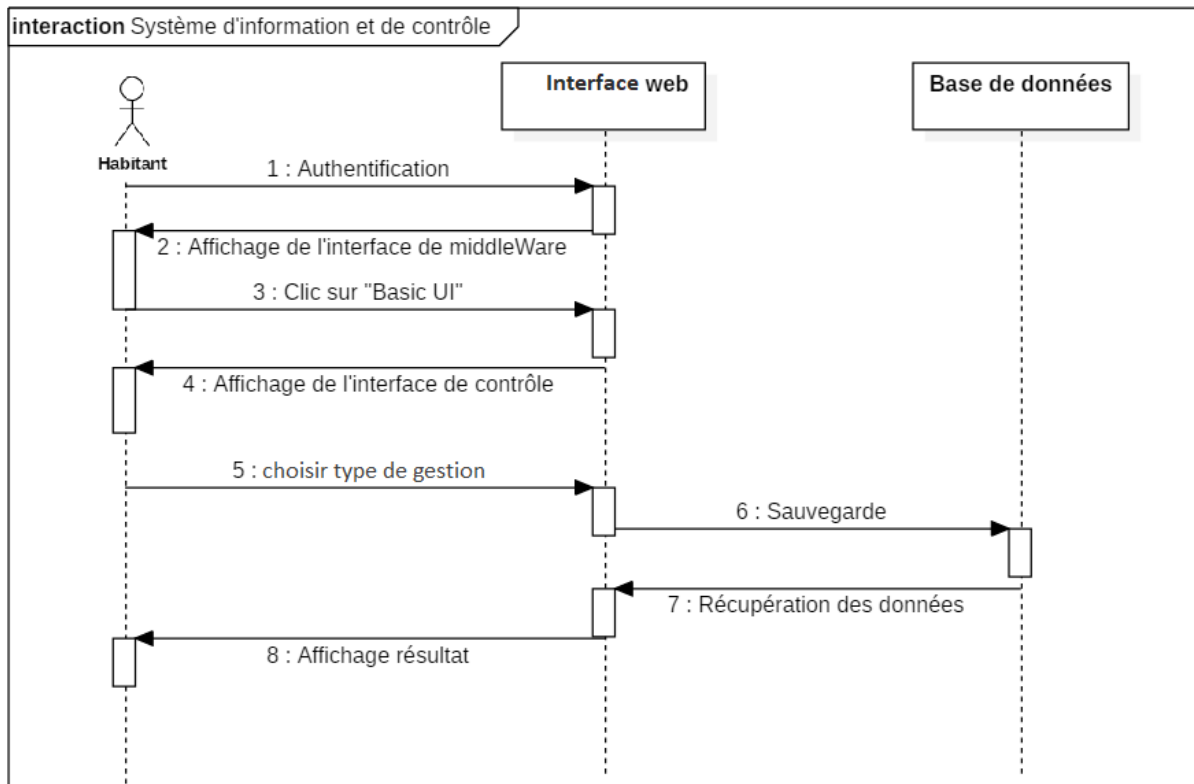


Figure 22 : Diagramme de séquence de l'application Web

### 1.3. Système automatique

Dans cette section, nous avons rédigé tous les scénarii possibles afin de permettre le système d'interagir automatiquement tout en informant l'habitant de l'état de sa maison.

Pour consulter ces scénarios, voir (Annexe D).

## 2. Diagramme de classe

Les diagrammes de classes permettant de décrire la structure des entités manipulées par les utilisateurs.

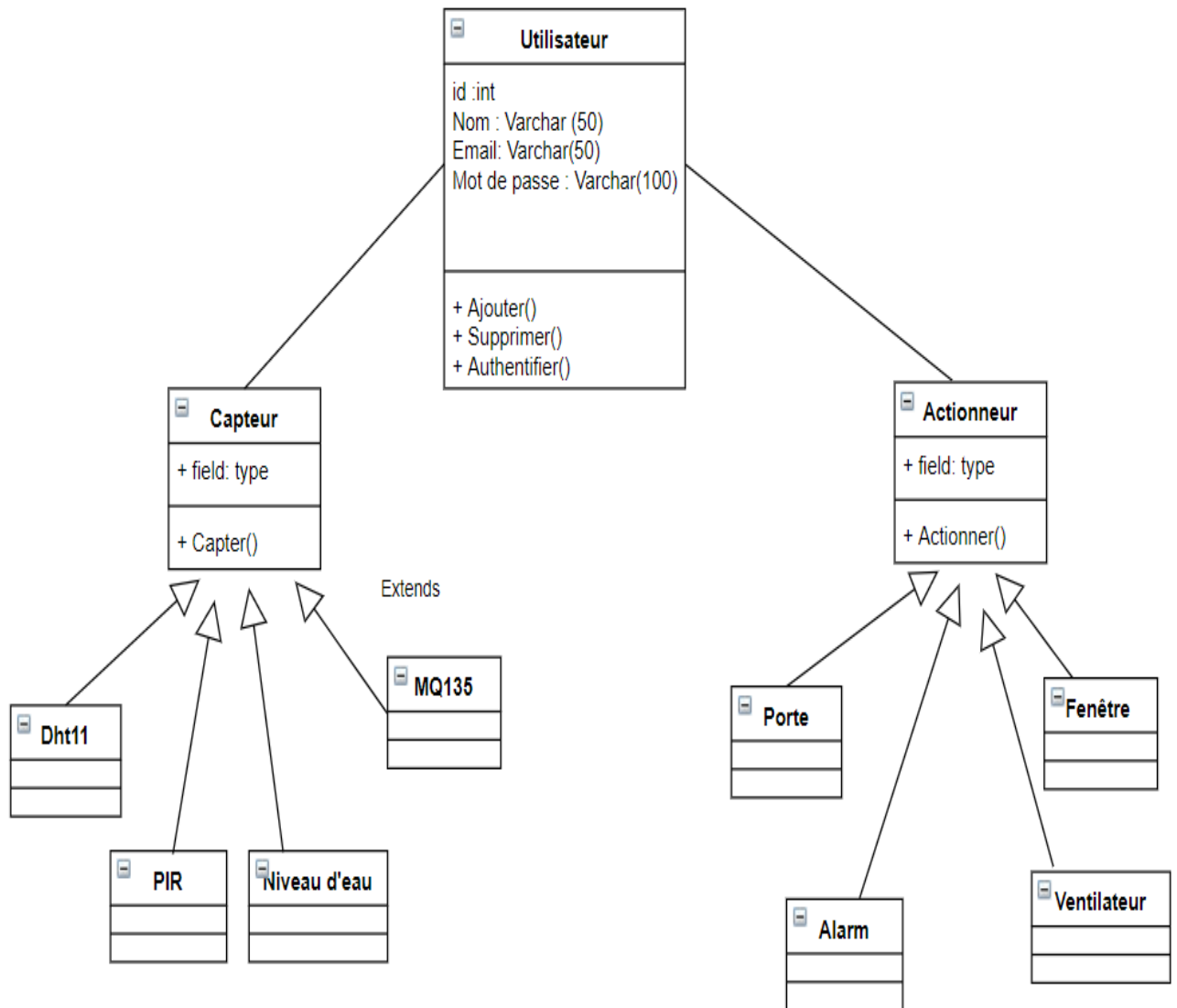


Figure 23 : Diagramme de classe

### 3. Diagramme d'activité

Les diagrammes d'activités permettent de mettre l'accent sur les traitements. Ils sont donc particulièrement adaptés à la modélisation du cheminement de flots de contrôle et de flots de données. En effet, ils représentent le flux de travail à partir d'un point de départ au point d'arrivée.

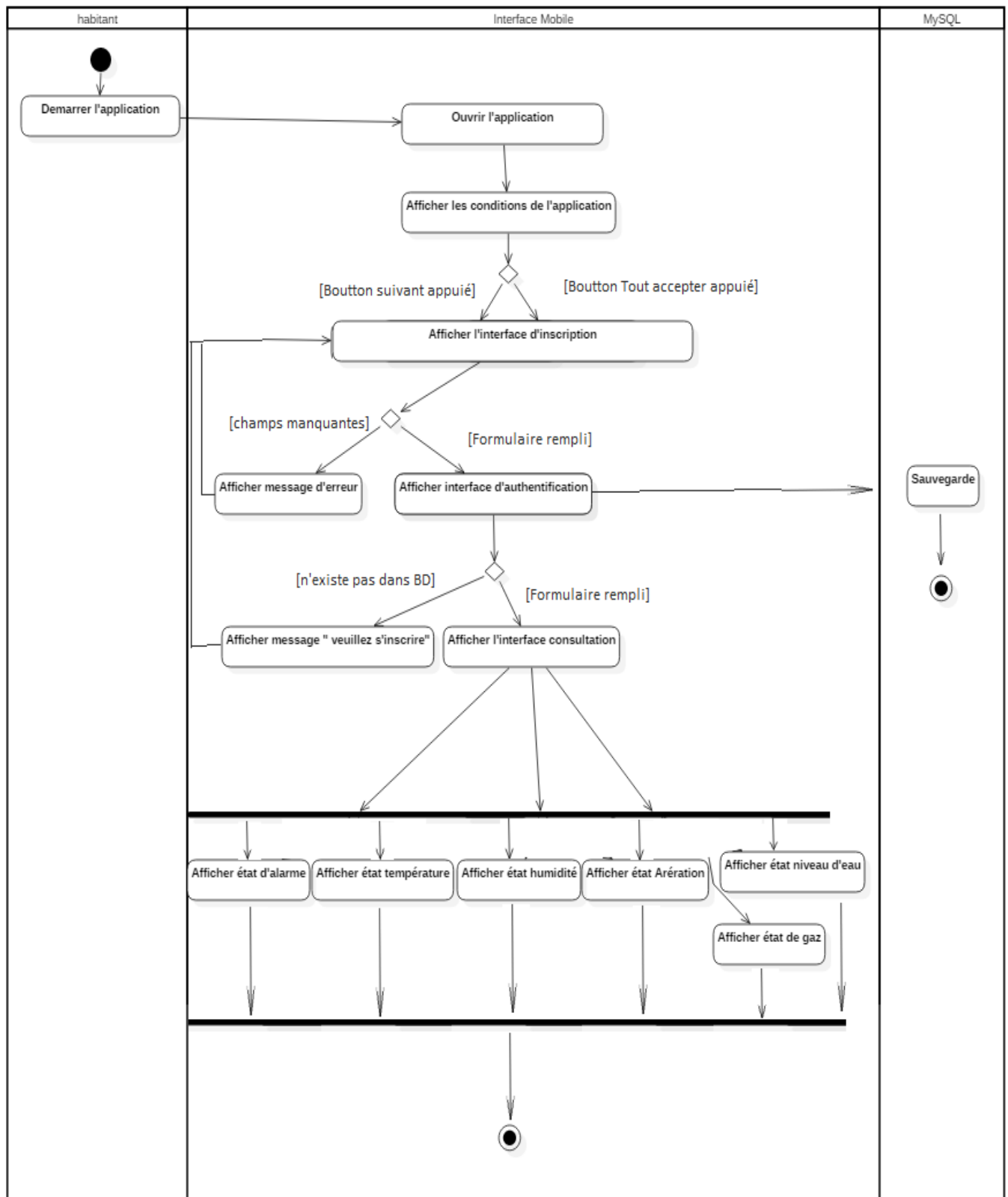


Figure 24 : Diagramme d'activité : « Consultation des informations »

## **Conclusion**

Dans ce chapitre nous avons présenté l'architecture globale de notre solution, par suite nous avons bien expliqué le fonctionnement de notre solution par des différents diagrammes à savoirs les diagrammes statiques et dynamiques. Maintenant, notre projet est prêt à être réalisé ainsi que les différents sous-systèmes de notre solution sont prêts à être codés.

Dans le chapitre suivant, nous entamerons les différentes phases et les étapes de la réalisation de notre projet.

# *Chapitre V :*

## *Réalisation*

## **Introduction**

Après avoir achevé l'étape de conception, nous allons entamer dans ce chapitre la partie réalisation.

D'abord, nous commençons par spécifier notre environnement matériel et logiciel. Ensuite, nous présentons à travers une figure de synthèse l'architecture générale de l'application. Enfin, nous exposons dans la dernière partie toutes les principales étapes que nous avons traversées pour la réalisation de ce projet.

### **I. Environnement de travail**

Dans cette partie nous avons étudié deux environnements de travail à savoir l'environnement matériel et l'environnement logiciel.

#### **1. Environnement matériel**

##### **1.1. Matériel de développement**

Pour la réalisation et l'implémentation des différentes parties de ce projet, nous avons utilisé deux ordinateurs qui représentent les caractéristiques suivantes :

##### Ordinateur1

- Marque : LENOVO,
- RAM : 12G,
- Disque dur : 2T,
- Processeur : Intel Core i5,
- Système d'exploitation : Windows 10.

##### Ordinateur2

- Marque : ASUS,
- RAM : 8 Go,
- Disque dur : 1T ,
- Processeur : Intel Core i5 ,
- Système d'exploitation : Windows 10 .

## 1.2. Matériel du nœud capteurs/actionneurs

Notre projet a fait appel à un matériel spécifique. Après avoir effectué une étude bien détaillée (voir chapitre3), le nœud de capteurs/actionneurs est assemblage entre la carte Arduino, les capteurs (dht11, MQ135, PIR, RFID) et les actionneurs (Buzzer, servomoteur, ventilateur) comme l'indique la figure ci-dessous :

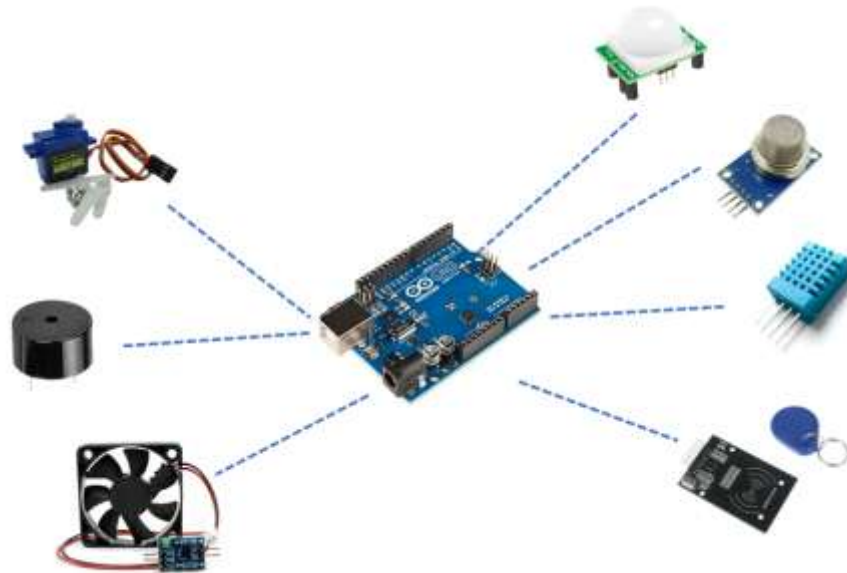


Figure 25 : Matériel du nœud capteurs/actionneurs

### 1.2.1. Arduino UNO

#### a. Description de la carte Arduino UNO

La carte Arduino UNO est une carte à microcontrôleur construite autour de l'ATmega328, Elle possède 14 broches d'entrée/sortie numériques (dont 6 peuvent servir de sorties normales, ou PWM), 6 entrées analogiques, un oscillateur à quartz de 16 MHz, un connecteur USB, un jack d'alimentation, une embase ICSP, et un bouton d'initialisation (reset).

#### b. Spécifications techniques de la carte Arduino UNO

Parmi les caractéristiques de la carte Arduino UNO nous citons :

**Alimentation : 5V**

**Fréquence : 16 MHz**

**Nombres de pins : 14 pins numériques et 6 pins analogiques**



**Courant de fonctionnement : 50 mA**

**Interface de communication : SPI, TWI**

### 1.2.2. Capteur dht11

#### a. Description

Le capteur numérique de température et d'humidité DHT11 est un capteur composite qui permet de donner un signal numérique en sortie de la température et de l'humidité. En utilisant l'acquisition exclusive des signaux numériques, il garantit une grande fiabilité et une excellente stabilité à long terme.

#### b. Spécifications techniques du capteur dht11

**Alimentation : 3.3-5.5V**

**Courant de fonctionnement : entre 0.3 mA et 2.5mA**

**Température mesurable : de 0°C à 50°C**

**Humidité mesurable : de 20% à 90%**

**Cout : 4.08€**

#### c. Branchement du capteur dht11

Dans la figure ci-dessous, on a présenté le branchement du capteur de température et humidité dht11 :

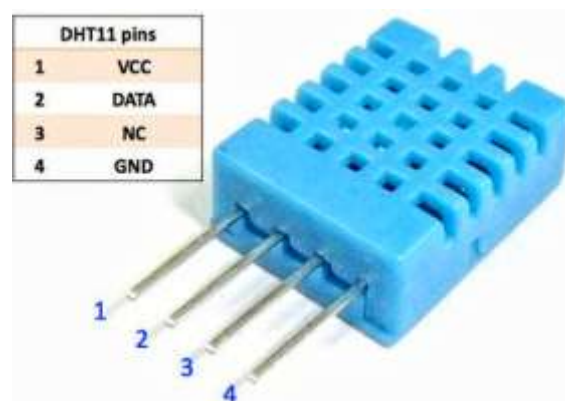


Figure 26 : Schéma de branchement du capteur dht11

On branche donc :

- Le VCC du dht11 sur le 5V,

- Le GRD du dht11 sur le GRD,
- Le pin NC est un pin inutilisable,
- Le pin DATA du dht11 sur n'importe quel pin de microcontrôleur.

### 1.2.3. Capteur détecteur de mouvement PIR

#### a. Description

Le capteur PIR permet de détecter des présences anormales dans un environnement. Ainsi, il perçoit les formes, les déplacements ou les volumes en utilisant la technique de l'infrarouge. Cet appareil a un rôle sécuritaire, et doit prémunir contre d'éventuels vols ou agressions. Le choix de son emplacement est vital. Son principe de fonctionnement est d'être sensible à la chaleur dégagée par un être vivant.

Le détecteur de mouvement transmet toutes les informations enregistrées en temps réel par un système radio permettant l'intervention rapide de la surveillance.

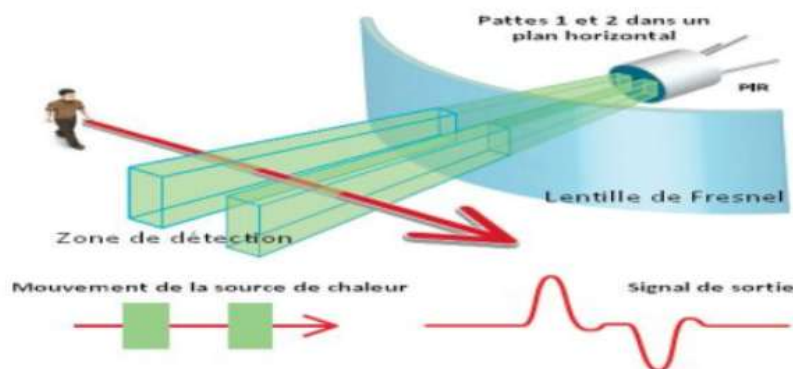


Figure 27 : Fonctionnement du capteur PIR

#### b. Spécifications techniques du capteur PIR

**Alimentation :** 5V – 20V

**Courant de fonctionnement :** 65mA

**Angle de détection :** 110°

**Poids :** 6g

**Cout :** 1.02 €

### c. Branchement du capteur

On branche donc :

- Le VCC du PIR sur le 5V,
- Le GRD du PIR sur le GRD,
- Le pin out sur le pin du microcontrôleur.



Figure 28 : Branchement du capteur PIR

### d. Réglage de la sensibilité du capteur

SX = Ajustement de la Sensibilité du capteur de 3-7m, visser pour augmenter,

TX = Ajustement du délai (Time) pendant lequel la sortie reste verrouillée sur HIGH après une détection de mouvement (TX). Visser pour augmenter la durée, jusqu'à 200 secondes.



Figure 29 : Le réglage de la sensibilité du capteur PIR

## 1.2.4. Capteur niveau d'eau

### a. Description

Un capteur de niveau est un dispositif électronique qui permet de mesurer la hauteur du matériau, en général du liquide, dans un réservoir ou un autre récipient.

### b. Spécifications techniques

Parmi les caractéristiques techniques du capteur niveau d'eau nous citons :

**Alimentation : 5V**

**Courant de fonctionnement : moins de 20 mA**

**Température de fonctionnement : de 10°C à 30°C**

**Humidité de fonctionnement : 10% à 90% sans condensation.**

**c. Branchement**



**Figure 30 : Branchement du capteur niveau d'eau ST045**

On branche donc :

- Le VCC du capteur sur le 5V,
- Le GRD du capteur sur le GRD,
- La branche Data sur le pin analogique de microcontrôleur.

### **1.2.5. Capteur MQ135**

**a. Description**

Le capteur de gaz MQ-135 est un capteur qui permet de mesurer la qualité de l'air. Il est utilisé dans les équipements de contrôle de la qualité de l'air pour bâtiments / bureaux, conviennent à la détection de l'ammoniac  $\text{NH}_3$ , l'oxyde d'azote ( $\text{NO}_x$ ), alcool, benzène, fumée et  $\text{CO}_2$ , avec une longue durée de vie et la stabilité fiable.

**b. Spécifications techniques**

Parmi les caractéristiques techniques du Buzzer nous citons :

**Alimentation : 5V**

**Courant de fonctionnement :** 0.15 A

**Type de gaz détecté :** Qualité d'air

**Puissance :** < 800mW

**Cout :** 9.33€.

### c. Branchement du capteur MQ135



**Figure 31 : Branchement du capteur gaz MQ135**

On branche donc :

- Le VCC du MQ135 sur le 5V,
- Le GRD du MQ135 sur le GRD,
- Le pin AOUT sur le pin analogique du microcontrôleur. Dans le cas où on va travailler avec le pin analogique,
- Le pin DOUT sur le pin numérique du microcontrôleur. Dans le cas où on va travailler avec le pin numérique.

## 1.2.6. Capteur RFID

### a. Description

Le module RFID-RC52216 est une carte d'interface compatible Arduino et Raspberry basé sur le circuit MFRC, il est principalement utilisé pour lire les badges RFID de type mifare pour permettre l'accès.

---

[16] : <https://www.gotronic.fr/pj2-sbc-rfid-rc522-fr-1665.pdf>

### b. Spécifications techniques

Parmi les caractéristiques techniques du module RFID nous citons :

**Alimentation** : 3,3 Vcc,

**Fréquence** : 13,56 MHz,

**Protocole** : Mifare,

**Interface** : SPI.

### c. Branchement

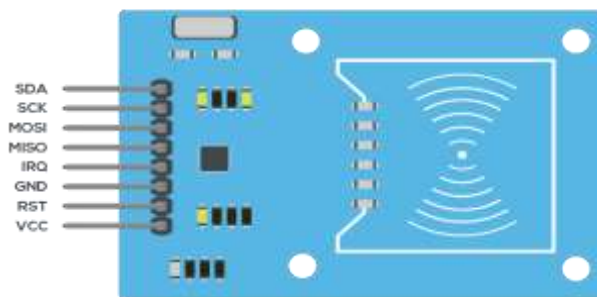


Figure 32: schéma de branchement du module RFID

On branche donc :

- Le VCC du module sur le 5V,
- Le GRD du module sur le GRD,
- Le RST du module sur une pin du microcontrôleur,
- Le MISO du module sur une pin du microcontrôleur,
- Le MOSI du module sur une pin du microcontrôleur,
- Le SCK du module sur une pin du microcontrôleur,
- Le SDA du module sur une pin du microcontrôleur.

## 1.2.7. Servomoteur

### a. Description

Un servomoteur est un moteur à courant continu, mais qui tourne suivant un angle varie entre 0° et 180°. Il est équipé d'un réducteur, son objectif est de réduire la vitesse,

d'augmenter le couple. Et d'un potentiomètre qui permet au servomoteur de garder l'angle choisit.

### **b. Spécifications techniques**

Parmi les caractéristiques d'un servomoteur nous citons :

**Alimentation** : 5V,

**Courant de fonctionnement** : 550 mA,

**Couples** : 2.5 kg/cm,

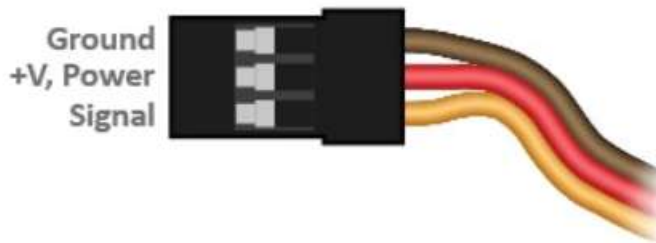
**Vitesse** : 0.1s / 60 °,

**Rotation** : de 0° à 180°,

**Poids** : 9g,

**Cout** : 2.32€.

### **c. Branchement du servomoteur**



**Figure 33 : Branchement du servomoteur**

On branche donc :

- Le VCC du servomoteur sur le 5V,
- Le GRD du servomoteur sur le GRD,
- La dernière branche (Signal) sur le pin de microcontrôleur.

### 1.2.8. Buzzer

#### a. Description

En cas d'intrusion (détection par détecteur de mouvement PIR, contact magnétique ou capteur infrarouge), le système va déclencher l'alarme par un Buzzer.

#### b. Spécifications techniques

Parmi les caractéristiques techniques du Buzzer nous citons :

**Alimentation** : 3.3 - 5.2V ;

**Courant de fonctionnement** : 25mA (max).

#### c. Branchement

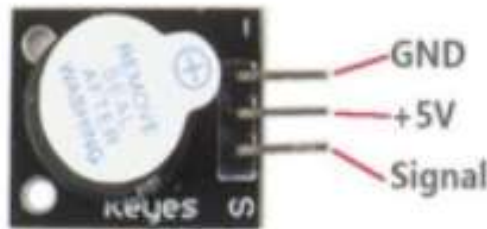


Figure 34 : Branchement du Buzzer

On branche donc :

- Le VCC du Buzzer sur le 5V,
- Le GRD du Buzzer sur le GRD,
- La dernière branche (Signal) sur le pin de microcontrôleur.

### 1.2.9. Ventilateur

#### a. Description

Le ventilateur permet la gestion d'aération, nous avons l'utilisé dans le but de la régulation de la température et de l'humidité ainsi que pour détruire toute sorte de fuit de gaz...

#### b. Spécifications techniques

Parmi les caractéristiques techniques du ventilateur nous citons :



**Alimentation : 12V,**

**Courant de fonctionnement : 50 mA.**

**c. Branchement**



**Figure 35 : Branchement du ventilateur**

On branche donc :

- Le VCC du ventilateur sur le 12V,
- Le GRD du ventilateur sur le GRD,
- La branche (Signal) sur le pin de microcontrôleur.

**1.3. Matériel du Gateway**

Concernant cette partie, nous avons choisi de travailler avec la carte Raspberry pi 3 B+.



**Figure 36 : Matériel du Gateway**

**a. Description de la carte Raspberry pi 3 B+**

Le Raspberry pi est un nano ordinateur de la taille d'une carte de crédit que l'on peut brancher à un écran et utilisé comme un ordinateur standard.

### **b. Spécifications techniques de la carte Raspberry pi 3 B+**

Parmi les caractéristiques de la carte Raspberry pi 3 B+ nous citons :

**Alimentation : 5V**

**Fréquence : 1.4GHz**

**Nombres de pins : 40 pins**

**Courant de fonctionnement : 50 mA**

**Interface de communication : SPI, TWI**

**Comporte : Wifi, Bluetooth 4.2**

**Vitesse d'Ethernet :300Mbps**

## **1.4. Matériel de communication**

### **a. Description du module ESP8266**

Ce module permet d'apporter une connectivité Wifi à un microcontrôleur. Nous l'avons utilisé pour assurer l'envoi et la réception des données entre le nœud de capteurs/actionneurs et le Gateway.

### **b. Spécifications techniques**

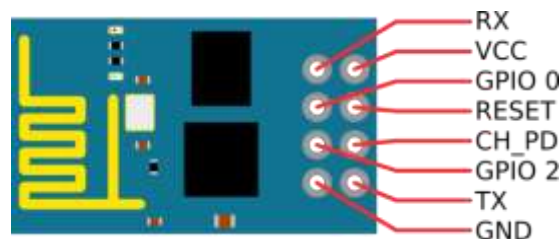
Parmi les caractéristiques techniques du module ESP8266 nous citons :

**Alimentation : 3.3V,**

**Porté : 250 m,**

**Cout : 8,60 €**

### **c. Branchement**



**Figure 37 : Branchement du module ESP8266**

On branche donc :

- Le VCC, RESET, CH\_PD de l'ESP sur le 3.3V,
- Le GRD de l'ESP sur le GRD,
- Le TX sur le RX de microcontrôleur,
- Le RX sur le TX de microcontrôleur.

## **2. Environnement logiciel**

Tout au long de cette section, nous présentons l'ensemble des outils logiciels utilisés pour le développement des différentes parties de notre solution.

### **2.1. Application embarquée**

#### **a. Système d'exploitation**



**Figure 38 : Logo de système d'exploitation "Raspbian"**

Pour faire fonctionner la carte Raspberry, elle suffit d'être configurée par un système d'exploitation.

En effet, nous avons choisi « Raspbian » qui représente en fait le système d'exploitation officiel pour tous les modèles de Raspberry pi.

#### **b. Logiciel de formatage de la carte SD**



**Figure 39 : Logo de logiciel de formatage de la carte SD "SDformatter"**

Avant d'importer le système d'exploitation dans la carte SD, nous avons formaté la carte SD avec le logiciel de formatage « SDformatter ».

**c. Logiciel de transfert**



**Figure 40 : Logo de logiciel de transfert SE "Win32 disk imager"**

Après le formatage de la carte SD, nous avons utilisé le logiciel « Win32 disk imager » pour transférer l'image disque du système d'exploitation « Raspbian » sur une carte mémoire SD.

**d. Logiciel de scan IP**



**Figure 41 : Logo de logiciel de scan IP "Advanced IP scanner"**

Afin de trouver l'adresse IP de la carte Raspberry pi, nous avons utilisé le logiciel « Advanced IP scanner ».

**e. Logiciel PuTTY**



**Figure 42 : Logo de logiciel "PuTTY"**

Pour accéder à la carte Raspberry, nous avons utilisé le logiciel « PuTTY » qui nous a permis la connexion en ssh sans n'nécessiter d'écran et clavier.

## **f. Logiciel OpenHab**



**Figure 43 : Logo de logiciel "OpenHab"**

Après avoir connecter à notre carte Raspberry pi, nous avons installé le logiciel « OpenHab » que nous avons choisi pour développer notre solution.

## **2.2. Application mobile**

### **a. Environnement de développement**



**Figure 44 : Logo de l'environnement de développement "Android Studio"**

« Android Studio » est un environnement de développement des applications Android basé sur IntelliJ IDEA. En outre, il fournit les outils les plus rapides pour créer des applications sur tous les types d'appareils Android.

## **2.3. Application Web**

### **a. Environnement de développement**



**Figure 45 : Logo de l'environnement de développement "Visual Studio Code"**

« Visual Studio Code » est un éditeur de code gratuit et open-source, il supporte plusieurs langages à savoir les langages de programmation, les langages de balisage, les langages de requête.

## b. Langages de développement



Figure 46 : logo PHP

- **PHP (Hypertext Preprocessor)** : est un langage de programmation libre, principalement utilisé pour produire des pages Web dynamiques via un serveur http.



Figure 47 : logo java

- **Java** : est un langage de programmation orienté objet.

## 2.4. Logiciel de modélisation

### a. StarUML



Figure 48 : Logo de logiciel de modélisation "StarUML"

« StarUML » est un logiciel de modélisation UML dont nous avons eu recours pour élaborer notre étude conceptuelle et pouvoir réaliser nos diagrammes.

## 2.5. Logiciel Sweet Home 3D



Figure 49 : logo de logiciel Sweet Home 3D

« Sweet Home 3D » est un logiciel open source dont nous avons l'utilisé pour préparer notre maquette.

## II. Architecture système

Rappelons que l'architecture générale de notre système est résumée dans la figure suivante :

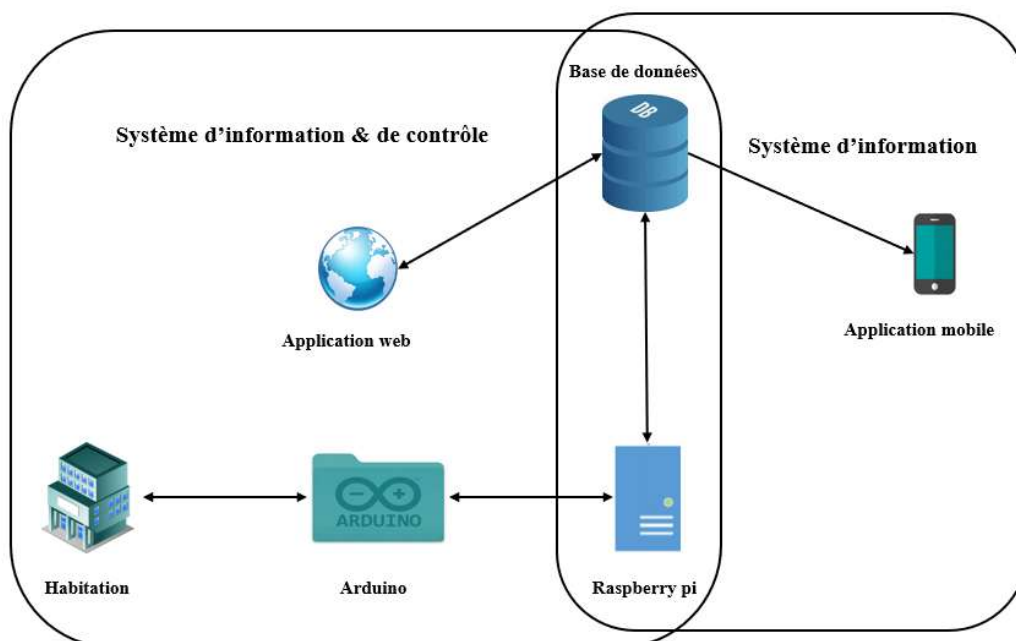


Figure 50 : Architecture système

### III. Travail réalisé : Partie software

#### 1. Application Mobile

##### 1.1. Acceptation des conditions de l'application

Avant d'accéder à l'application mobile de notre solution, l'utilisateur doit passer par l'étape d'acceptation des conditions de l'application.



Figure 51 : Interface d'acceptation des conditions de l'application mobile

##### 1.2. Création du compte

Après avoir accepté les conditions de l'application, l'utilisateur doit créer un compte pour qu'il soit sauvegarder dans la base de données.





Figure 52 : Interface d'inscription

### 1.3. Authentification

Après avoir créé un compte, cette interface permet à l'utilisateur d'introduire son adresse mail et son mot de passe afin d'accéder à l'interface d'informations.



Figure 53 : Interface d'authentification

### 1.4. Interface de consultation

En appuyant sur le bouton « login » de « l'interface d'authentification », l'utilisateur va être capable de consulter toute sorte d'informations sur l'état de sa maison à distance.

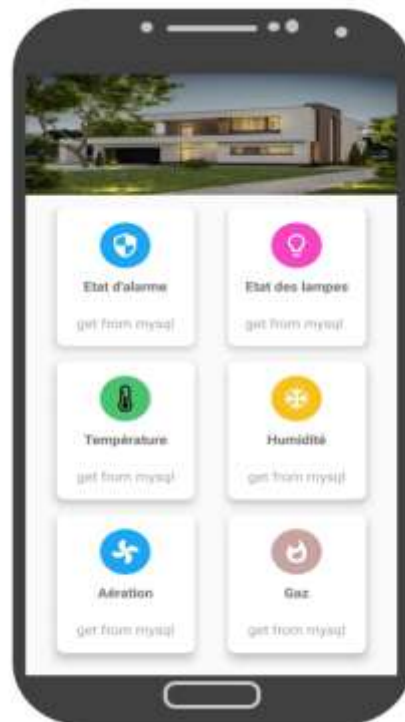
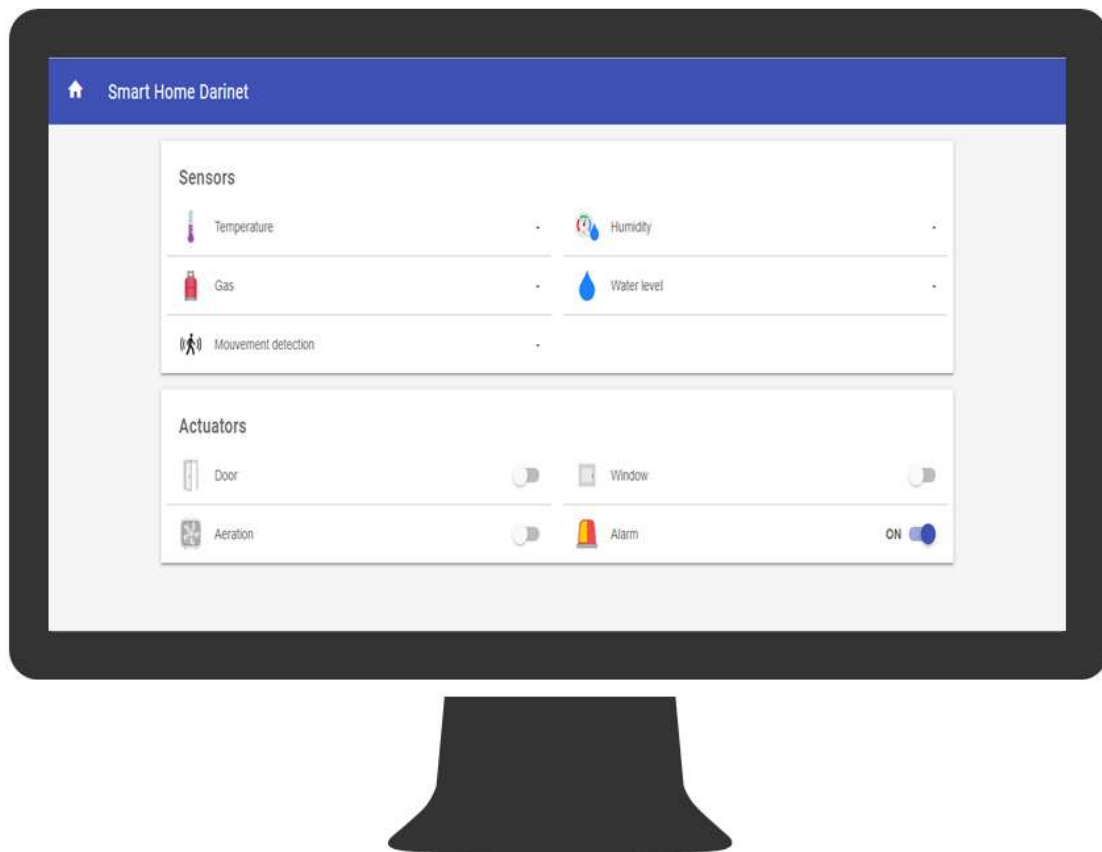


Figure 54 : Interface de consultation des informations

## 2. Application web

L'application web dans notre projet offre un système d'information et de contrôle à l'habitant pour gérer et contrôler sa maison à distance ainsi que consulter des informations concernant son état.



## IV. Travail réalisé : Partie hardware

### 1. Câblage du nœud de capteurs/actionneurs

Dans cette section, nous avons présenté la partie nœuds de capteurs/actionneurs avec un câblage très clair et lisible afin de se rappeler après le câblage de chaque composant utilisé.

La figure qui suit, présente ce câblage :

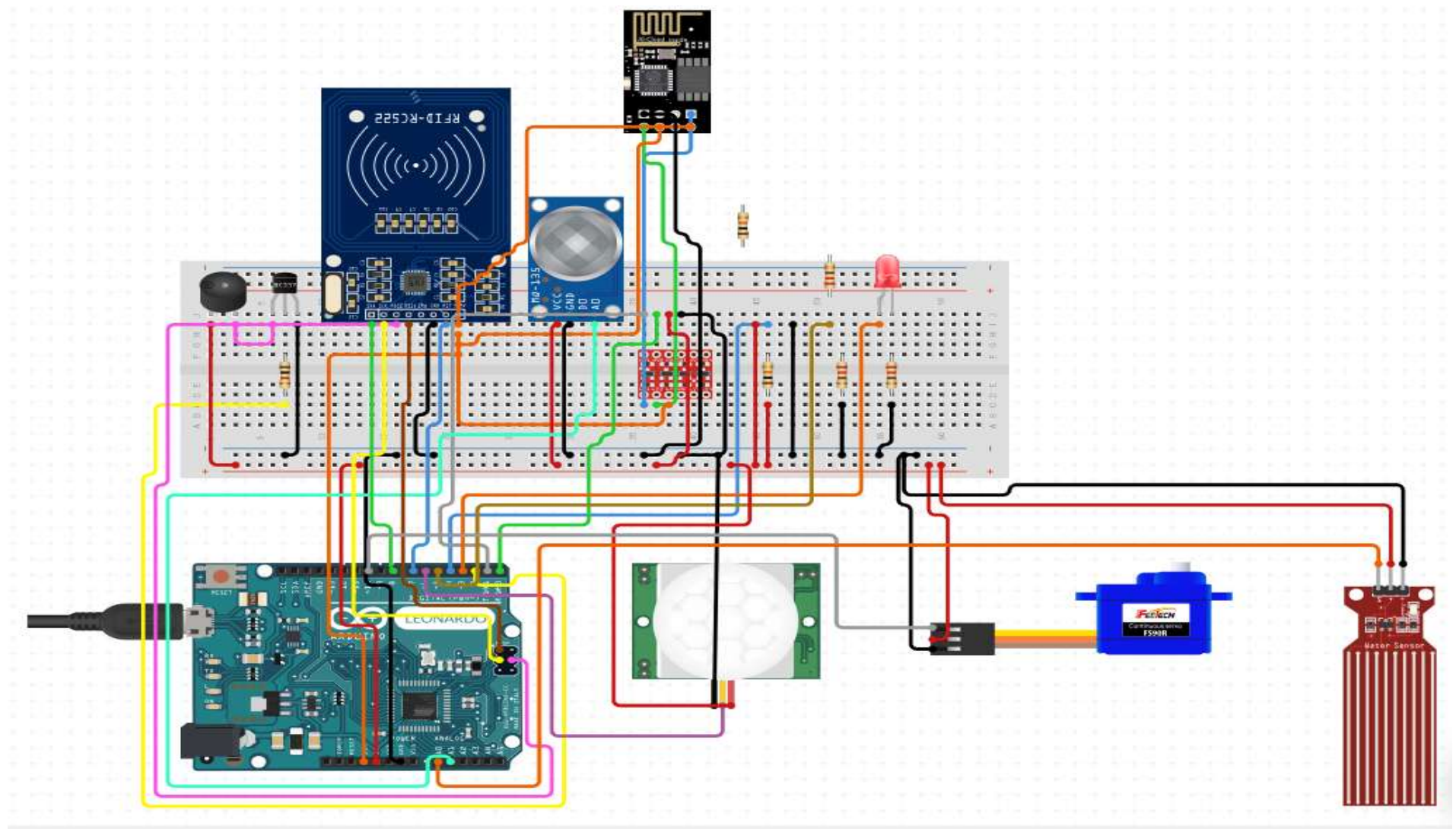
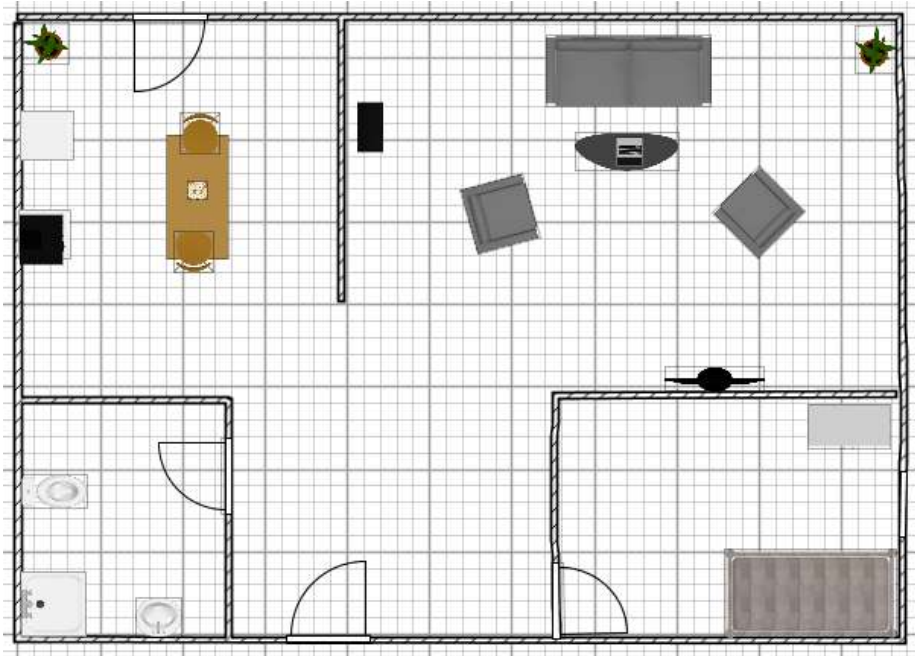


Figure 55 : schéma de câblage du nœud de capteurs/actionneurs

## 2. Maquette

La réalisation d'une maquette pour n'importe quel projet est une phase très importante. C'est pour cette raison, nous avons préparé une maquette sous forme d'une maison, afin d'intégrer notre travail dedans. Voir les deux figures ci-dessous :



**Figure 56 : Maquette 2D**



**Figure 57 : Maquette 3D**

## **Conclusion**

Tout le long de ce chapitre nous avons présenté en premier lieu les matériels et les logiciels que nous avons utilisé dans notre projet. Ensuite, nous avons fait un rappelle à l'architecture de notre solution et nous avons fini par présenter le travail réaliser côté matériel (SW) et cité software (HW).

# Conclusion et perspectives

Tout au long de notre stage de fin d'étude, nous avons été amenées à concevoir et réaliser un système connecté permettant le contrôle environnemental des maisons à partir d'une application mobile ou encore à partir d'une application web. La réalisation du projet s'est étalée sur une période d'environ quatre mois au sein de l'entreprise « EXSOL SYSTEMS ».

Ce stage nous a offert une occasion profitable pour appliquer nos connaissances et surtout en ajouter pleins d'autres. Nous sommes intégrés dans une très bonne équipe et dans un environnement de recherches et de nouveauté.

En fait, notre projet de fin d'études consiste à développer au premier lieu une solution domotique temps réel. En deuxième lieu, un système d'information (Application mobile) dans le but d'informer l'habitant de l'état de sa maison, ainsi qu'un système de contrôle et d'information (Application web) qui permet à l'habitant de gérer et de contrôler sa maison à distance en cas d'urgence.

Le point de départ de la réalisation de ce projet était une récolte des informations nécessaires.

Par la suite, nous nous sommes intéressées à l'analyse et spécification des besoins qui nous a permis d'extraire les grands axes de travail ainsi que distinguer les différents acteurs interagissant avec notre solution.

En outre, la conception nous a permis de clarifier et mettre en évidence les interactions entre les sous-systèmes de notre solution. Vers la fin de cette étape la structure globale de l'application a été fixée.

Le dernier volet était la partie réalisation et implémentation où nous avons atteint et accompli les objectifs de notre projet de fin d'étude.

En termes de perspectives, notre projet de fin d'études est également extensible. Il peut avoir plusieurs ajouts et améliorations.

D'ailleurs, nous allons ajouter des caméras à notre solution afin de filmer et contrôler tout l'espace de l'habitation. Et cette amélioration sera faite dans le but d'offrir au client la possibilité de visualiser ce qui se passe à l'intérieure de sa maison en temps réel.

C'est pour cette raison qu'on a choisi de travailler avec une communication wifi et non pas avec une communication radio.



# Nétographie

[1] : <http://exsolsystems.tn/>

[2] : <https://www.jeedom.com/site/fr/>

[3] : <http://www.domoticz.com>

[4] : <https://www.home-assistant.io>

[5] : <https://www.openhab.org>

[6] : <https://www.heflo.com/fr/blog/agile/methode-agile-scrum/>

[7] : <https://www.26academy.com/cours/les-methodes-claÖ-methodes-agiles/>

[8] : <https://www.electroschematics.com/6393/lm35-datasheet/>

[9] : <https://www.detecteur-mouvement.com/>

[10] : <https://www.gotronic.fr/cat-niveau-1136.htm>

[11] : <https://www.gotronic.fr/pj2-sbc-rfid-rc522-fr-1665.pdf>

[12] : <https://components101.com/microcontrollers/raspberry-pi-3-pinout-features-datasheet>

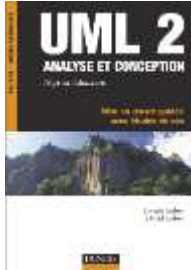
[13] : <https://www.farnell.com/datasheets/1682209.pdf>

[14] : <http://www.mantech.co.za/datasheets/products/A000047.pdf>

[15] : <http://www.farnell.com/datasheets/1682238.pdf>

[16] : <https://components101.com/microcontrollers/beaglebone-black-pinout-datasheet>

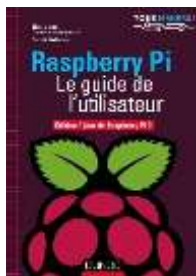
# Bibliographie



Gabay, Joseph. Gabay, David. (2008). Analyse et conception. Paris, France : DUNOD.



Schwartz, M. (2011). Raspberry Pi et l'ESP 8266 pour la domotique. Paris, France : DUNOD.



Eben, Upton. Gareth, Halfacree. (2010). Raspberry Pi « le guide de l'utilisateur ». Paris, France : DUNOD.

# **Annexes**

## Annexe A : Etude des middlewares existants

	Domoticz	OpenHab	OpenRemote	Home Assistant
<b>License</b>	Open Source	Open Source	Open Source	Open Source
<b>Compatibilité système</b>	<ul style="list-style-type: none"> <li>- Linux</li> <li>- Windows</li> <li>- Raspbian</li> </ul>	<ul style="list-style-type: none"> <li>- Linux</li> <li>- Windows</li> <li>- MacOS</li> <li>- Raspbian</li> </ul>	<ul style="list-style-type: none"> <li>- Android</li> <li>- iOS</li> <li>- Raspbian</li> </ul>	<ul style="list-style-type: none"> <li>- Linux</li> <li>- Windows</li> <li>- MacOS</li> <li>- Raspbian</li> </ul>
<b>Compatibilité technologique</b>	<ul style="list-style-type: none"> <li>- Z-wave</li> <li>- Rfxcom</li> <li>- En-Ocean</li> <li>- Zigbee</li> </ul>	<ul style="list-style-type: none"> <li>- MQTT</li> <li>- TCP/UDP</li> <li>- Z-wave</li> <li>- Zigbee</li> <li>- KNX</li> </ul>	<ul style="list-style-type: none"> <li>- En-Ocean</li> <li>- TCP/IP</li> <li>- ModBus</li> <li>- SNMP</li> <li>- Z-wave</li> <li>- UDP</li> <li>- KNX</li> </ul>	<ul style="list-style-type: none"> <li>- MQTT</li> <li>- TCP/IP</li> </ul>
<b>Flexibilité</b>	<ul style="list-style-type: none"> <li>- Plateforme très stable,</li> <li>- Il est assez limité en termes de périphériques et configuration.</li> </ul>	<ul style="list-style-type: none"> <li>- N'est pas le plus simple</li> <li>- Flexible</li> </ul>	<ul style="list-style-type: none"> <li>- Flexible</li> </ul>	<ul style="list-style-type: none"> <li>- Pas de flexibilité</li> </ul>
<b>Communauté</b>	<ul style="list-style-type: none"> <li>- Il compte moins d'utilisateur que les autres plateformes,</li> <li>- Il a une communauté moyenne</li> </ul>	<ul style="list-style-type: none"> <li>- Les personnes les bien informées dans les forums sont tjrs disponible pour répondre aux questions,</li> <li>- Il a une large communauté.</li> </ul>	<ul style="list-style-type: none"> <li>- Il a une moyenne communauté.</li> </ul>	<ul style="list-style-type: none"> <li>- La communauté se développe très rapidement.</li> </ul>

## Annexe B : Technologies de communications

	Wifi			Zigbee		NFC	Bluetooth	WiMax		Ultra Wide Band UWB	RFID	LoRaWan	Z-Wave	
Spécification	IEEE 802.11b	IEEE 802.11a	IEEE 802.11g	IEEE 802.15.4	IEEE 802.15.4a	ISO/IEC 14443 et ISO/IEC 18000-3	IEEE 802.15.1	IEEE 802.16	IEEE 802.16a	IEEE 802.15.3	EPC 1.0 - ISO 10536 et ISO 14443	IEEE 802.20	Z-Wave Alliance ZAD12837/ITU -T G.9959	
Fréquence	2.4 GHz	5.5 GHz	2.4 GHz	2,4 GHz	900 KHz	13,56 MHz	2 à 11 GHz	10 et 66 GHz	2 à 11 GHz	2.4 GHz	13,56 Mhz	868 MHz	900 MHz	
Portée intérieur	150 m	10 m	27 m	10m	75m	10 cm	Bth4 : 10 m Bth5 : 40 m	50 km		Dizaines de mètres	3m	2Kms	>40 m	
Portée extérieur	500 m	35 m	75 m	400m			Bth4 : 50 m Bth5 : 200 m					5 kms	>100m	
Taux de transfert	11Mbps	54 Mbps		250 Kb/s	20Kb/s	424 kbps	Bth4: 2Mbps Bth5: 1Mbps	70Mbps/s		Centaines de Mb/s	10Mbit / s	0,3 kbps à 50 kbps	100 Kbit/s	
Consommation	Faible			Faible		Faible	Faible	Très faible		Faible	Faible	Faible	Faible	
Technologie de réseau	IEEE	IEEE	IEEE	IEEE	IEEE	ISO	Scatternet	IEEE	IEEE	IEEE	ISO	IEEE		
Références	<a href="https://fr.scribd.com/document/250491591/Les-differentes-technologies-sans-fil-pdf">https://fr.scribd.com/document/250491591/Les-differentes-technologies-sans-fil-pdf</a> <a href="https://www.commentcamarche.net/contents/1280-wifi-portee-et-debit">https://www.commentcamarche.net/contents/1280-wifi-portee-et-debit</a>					<a href="https://www.digikey.fr/fr/articles/techzone/2017/oct/comparing-low-power-wireless-technologies">https://www.digikey.fr/fr/articles/techzone/2017/oct/comparing-low-power-wireless-technologies</a>	<a href="https://www.amarinfotech.com/differences-comparisons-bluetooth-5-vs-4-2.html">https://www.amarinfotech.com/differences-comparisons-bluetooth-5-vs-4-2.html</a> <a href="https://fr.scribd.com/document/250491591/Les-differentes-technologies-sans-fil-pdf">https://fr.scribd.com/document/250491591/Les-differentes-technologies-sans-fil-pdf</a> <a href="http://www.hobbytronics.co.uk/mfrc522-reader">http://www.hobbytronics.co.uk/mfrc522-reader</a>					<a href="https://fr.rs-online.com/web/generalDisplay.html?id=v/loronline.com/web/generalDisplay.html?id=v/loronline.com/stylisme.com/zigbee-ou-z-wave-lequel-choisir/">https://fr.rs-online.com/web/generalDisplay.html?id=v/loronline.com/stylisme.com/zigbee-ou-z-wave-lequel-choisir/</a> <a href="https://www.silabs.com/documents/login/datasheets/DSH12435-15.pdf">https://www.silabs.com/documents/login/datasheets/DSH12435-15.pdf</a> <a href="https://www.silabs.com/documents/login/datasheets/DSH12436-Datasheet-ZD85202-Z-Wave-Development-Board.pdf">https://www.silabs.com/documents/login/datasheets/DSH12436-Datasheet-ZD85202-Z-Wave-Development-Board.pdf</a>		

# Annexe C : Configuration du Gateway

## I. Système d'exploitation

### 1. Introduction

Avant de brancher la carte Raspberry pour débiter la configuration, nous devons d'abord installer un système d'exploitation sur la carte SD. Pour ce faire, nous allons télécharger Raspbian depuis le site officiel du Raspberry pi. En fait, Raspbian est un **système d'exploitation libre** et gratuit basé sur **Debian** optimisé pour fonctionner sur un **Raspberry Pi**.

### 2. Téléchargement de Raspbian

En ce qui se suit les principales étapes de téléchargement du système sur la carte SD :

- Télécharger Raspbian depuis le site officiel du Raspberry pi,

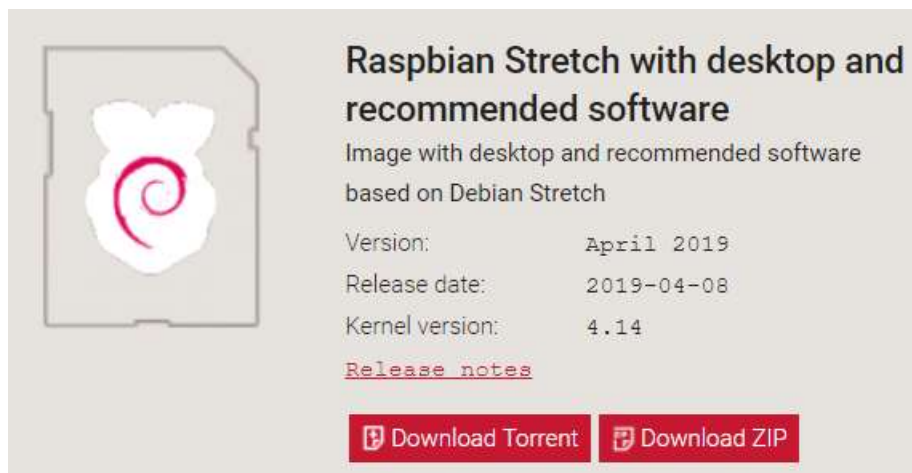
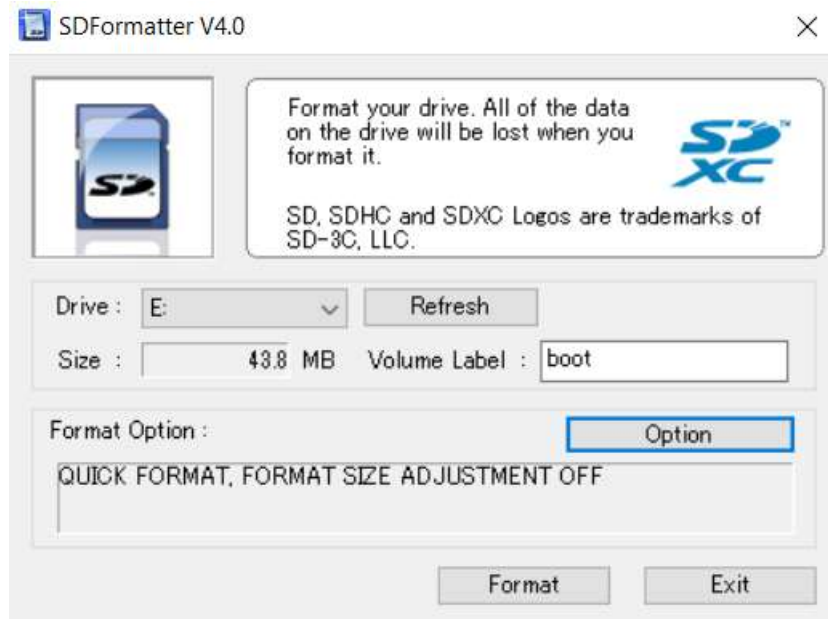


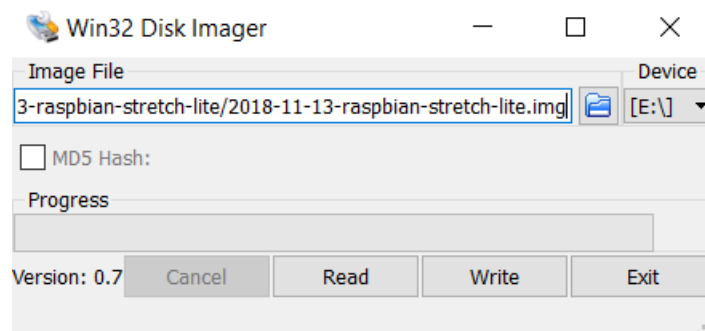
Figure 58 : installation système d'exploitation Raspbian

- Installer le logiciel SD Formatter pour le formatage de la carte SD,



**Figure 59 : formatage de la carte SD**

- Utiliser le logiciel win32 disk imager pour importer l'image du système d'exploitation Raspbian,



Une fois l'importation est terminée, il ne reste qu'à connecter la carte SD dans le port micro SD de la Raspberry pi ainsi que brancher cette dernière pour l'installation de Raspbian.

- Chercher l'@ip de notre carte Raspberry en utilisant Advanced IP Scanner
- Taper l'adresse ip de notre carte Raspberry dans le logiciel PuTTY pour terminer l'installation de notre système d'exploitation.

## **II. Installation et configuration des bibliothèques**

### **1. Openhab2**

Les étapes et les instructions d'installation et de configuration de middleware openhab2 sont les suivantes :

#### **Etape1 : Installation des dépendances**

- Mettre à jour tous les paquets existants :

```
$ sudo apt-get update && sudo apt-get upgrade
```

- Installer le protocole de transmission de données :

```
$ wget -qO - https://bintray.com/user/downloadSubjectPublicKey?username=openhab' |  
sudo apt-key add -
```

```
$ sudo apt-get install apt-transport-https
```

- Mettre à jour tous les paquets existants :

```
$ sudo apt-get update
```

#### **Etape 2 : installation du middleware openhab2**

2. Après avoir installé le protocole https de transmission de données, nous avons installé openhab2 :

```
$ sudo apt-get install openhab2
```

```
$ sudo apt-get install openhab2-addons
```

```
$ sudo systemctl start openhab2.service
```

```
$ sudo systemctl status openhab2.service
```

```
$ sudo systemctl daemon-reload
```

```
$ sudo systemctl enable openhab2.service
```

#### **Etape 3 : Partage réseau**

3. Pour pouvoir accéder aux fichiers openhab2 depuis le pc il suffit d'installer les partages réseau Samba :

```
$ sudo apt-get install samba samba-common-bin
```

```
$ sudo nano /etc/samba/smb.conf
```



4. Accéder au fichier smb.conf et activer le support WINS :

```
$ sudo nano /etc/samba/smb.conf
```

5. Modifier : wins support = yes

6. Ajouter le contenu ci-dessous dans le fichier smb.conf :

```
[openHAB2-userdata]
```

```
Comment=openHAB2 userdata
```

```
Path=/var/lib/openhab2
```

```
Browseable=Yes
```

```
Writeable=Yes
```

```
Only guest=no
```

```
Public=no
```

```
Create mask=0777
```

```
Directory mask=0777
```

```
[OpenHAB2-conf]
```

```
Comment=openHAB2 site configuration
```

```
path=/etc/openhab2
```

```
browseable=Yes
```

```
writeable=Yes
```

```
Only guest=no
```

```
Public=no
```

```
Create mask=0777
```

```
Directory mask=0777
```

```
[openHAB2-logs]  
comment=openHAB2 logs  
path=/var/log/openhab2  
browseable=Yes  
writeable=Yes  
only guest=no  
Public=no  
Create mask=0777  
Directory mask=0777
```

Donner un mot de passe pour accéder aux fichiers openhab2 :

```
$ sudo smbpasswd -a openhab
```

#### **Etape 4 : Installation mosquito**

```
$ sudo wget http://repo.mosquitto.org/debian/mosquitto-repo.gpg.key  
$ sudo apt-key add mosquitto-repo.gpg.key  
$ cd /etc/apt/sources.list.d/  
$ sudo wget http://repo.mosquitto.org/debian/mosquitto-stretch.list  
  
$ sudo apt-get update  
  
$ sudo apt-get install mosquitto mosquitto-clients
```

## **Annexe D : Scénarios**

### **Scénario détection de gaz**

- S'il y'a une détection de gaz, le système ouvre le volet et envoi un mail à l'habitant.
- S'il y'a un fuit de gaz, le système ouvre le volet, met en marche le système d'aération (Ventilation) envoi un mail et un sms à l'habitant. Scénario température du matin,
- Si la température détectée par le capteur dht11 entre 20 °C et 25 °C ( $20 < xT < 25$ ) et la date d e système est supérieure à 18h, le système ferme le volet, ouvre le système d'aération (ventilateur) et envoi un mail à l'habitant,
- Si la température est supérieure ou égale à 26 °C ( $xT \geq 26$  °C), le système met en marche une alarme, ouvre le volet et la porte, met en marche le système d'aération, et envoi un mail et un sms à l'habitant.

### **Scénario température de la nuit**

- Si la température détectée par le capteur dht11 est supérieure à 20 °C, le système met en marche une alarme, ouvre le volet et la porte, met en marche le système d'aération, et envoi un mail et un sms à l'habitant.

### **Scénario humidité**

- S'il y'a une détection d'humidité, le système ouvre le volet et la porte entre 10h et 12h, et envoi un mail et un sms à l'habitant. Scénario température et humidité
- S'il y'a une détection d'humidité, le système ouvre le volet et la porte entre 10h et 12h, et envoi un mail et un sms à l'habitant.
- Si la température détectée par le capteur dht11 entre 20 °C et 25 °C ( $20 < xT < 25$ ) et la date d e système est supérieure à 18h, le système ferme le volet, ouvre le système d'aération (ventilateur) et envoi un mail à l'habitant.
- Si la température est supérieure ou égale à 26 °C ( $xT \geq 26$  °C), le système met en marche une alarme, ouvre le volet et la porte, met en marche le système d'aération, et envoi un mail et un sms à l'habitant. Scénario détection de mouvement

- S'il y'a une détection de mouvement dans la maison, le système ferme le volet et le porte, met en marche l'alarme, et envoi un mail et un sms pour notifier l'habitant.

Scénario fuit d'eau

- S'il y'a un fuit d'eau dans la maison, le système envoi un mail et un sms pour notifier l'habitant.

### **Scénario de luminosité**

- A 18h, le système allume les lampes du salon.

## **Annexe E : Configuration de l'application Android**

### **I. Installation des bibliothèques**

Implementation 'com.android.support:cardview-v7:27.0.0'

Cette bibliothèque ajoute un support pour le CardView widget, qui permet d'afficher des informations à l'intérieur des cartes qui ont un aspect cohérent sur n'importe quelle application.

Implementation 'com.android.support:design:27.1.0'

Bibliothèque de support permettant de réaliser facilement une belle application, en embarquant une bonne partie des nouveaux composants et layouts qui font l'identité du Material Design, tel que le Floating Action Button ou les Collapsing toolbar

Implementation 'com.android.volley: volley:1.1.0'

Volley est une bibliothèque HTTP qui facilite la mise en réseau d'applications Android, nous avons utilisé cette bibliothèque pour connecter à la base de données.

### **II. Ajout permission**

Notre application nécessite une connexion internet afin de pouvoir consulter les informations concernant la maison de l'habitant :

```
<uses-permission android:name="android.permission.INTERNET" />
```

## تلخيص

الهدف الرئيسي لمشروعنا النهائي للدراسات هو تطوير نظام الأتمتة المنزلية يسمح بالتحكم عن بعد في المنازل. يتطلب تطوير نظامنا استعمال المستشعرات لتتبع الأحداث الخارجية وذلك من أجل التحكم في المشغلات باستعمال بوابة، مزودة ببرنامج وسيط.

كما يمكن نظامنا حولا للتحكم باستعمال الجوال أو الويب.

**الكلمات المفاتيح:** حل الأتمتة المنزلية، بوابة، برنامج وسيط.

---

## Résumé

L'objectif principal de notre projet de fin d'études consiste à développer une solution domotique permet le contrôle environnemental des maisons.

Le développement de notre solution devra permettre la communication entre des capteurs et des événements extérieurs produites afin de mettre en marche des actionneurs passant par un Gateway, qui est équipé d'un middleware.

Cette solution peut également être contrôlée par mobile ainsi que par web.

**Mots clés :** Solution domotique, Gateway, middleware.

---

## Abstract

The main goal of our graduation project is to develop a home automation solution that allows the environmental control of homes.

The development of our solution will have to allow the communication between sensors and external events produced in order to start actuators passing through a Gateway, which is equipped with a middleware.

This solution can also be controlled by mobile as well as by web.

**Keywords:** Home automation solution, Gateway, middleware.