

13/10/2017

Délivrable n° 1

Projet ZenBox

« J'atteste que ce travail est original, qu'il indique de façon appropriée tous les emprunts, et qu'il fait référence de façon appropriée à chaque source utilisée »

Léo Guilpain & François Isabelle & Max Aguirre

Table des matières

1. Présentation de l'équipe	3
2. Bilan de compétences	3
2.1 Compétences globales.....	3
2.2. Compétences en programmation	4
3. Affectation des tâches et structure de l'équipe	4
4. Présentation de l'objectif	5
5. Introduction au cahier des charges	5
6. Mode de fonctionnement	10
6.1. Au sein de l'équipe	10
6.2. Tâches à accomplir	10
7. Identification des risques	12
8. État de l'art	12
8.1. Différentes Box domotiques.....	12
8.1.1. Jeedom Smart.....	12
8.1.2. Eedomus+	14
8.1.3. Home center 2.....	15
8.1.4. Somfy Box.....	17
8.2. Outils d'assistances vocales.....	18
8.2.1. Amazon Echo	18
8.2.2. Google Home.....	19
8.2.3. HomePod	20
8.3. Différents logiciels domotiques.....	20
8.3.1. Jeedom Interaction.....	20
8.3.2. Domoticz.....	21
8.3.3. Yadoms	22
8.3.4. Calaos	23
8.3.5. Home Assistant.....	23
8.3.6. OpenHab.....	24
8.3.7. TaHoma	25
8.3.8. Fibaro.....	25
8.4. Réseau de neurones	26
8.5. Logiciels développant les réseaux de neurones	26
8.6. Différents protocoles de transmissions existants	28
8.6.1. Zwave.....	28
8.6.2. RFXcom	28

8.6.3. RFLink	29
8.6.4. EnOcean.....	29
8.6.5. xPL	29
8.6.6. Zigbee	29
8.6.7. IO Homecontrol	30
8.6.8. RTS (Radio Technology Somfy)	30
8.7. Les équipements de la maison	30
9. Conclusion	33
Bibliographie.....	34
Annexe :.....	36

1. Présentation de l'équipe

L'équipe se compose des élèves ingénieurs Max Aguirre (chef de projet), François Isabelle et Léo Guilpain. Nous étudions tous les trois en ESIR2 et nous suivons la spécialité Internet des Objets, Sécurité et villes intelligentes.

2. Bilan de compétences

Ce bilan de compétences va nous permettre de se répartir les différentes tâches et surtout d'affecter à chacun un rôle bien précis et ainsi d'être plus efficaces.

Pour faire ce bilan, nous avons établi un code.

Base 1 à 5 :

1 : Jamais vu / Pas maîtrisé

2 : Vu / Connaissances rudimentaires

3 : Vu / Un peu maîtrisé

4 : Vu / Moyennement maîtrisé

5 : Vu / Bien maîtrisé

2.1 Compétences globales

	Électronique	Réseau	Transmission
Léo	4	2	2
François	3	2	2
Max	4	2	3

Tableau 1 : Compétences globales

Grâce à ce tableau (Tableau 1) on peut voir clairement toutes les compétences de chacun et ceci dans les différents domaines.

Pour l'électronique, Max est celui qui possède le plus de compétences même si Léo et François possèdent également des compétences et seront donc capables d'apporter leurs savoirs en cas de problèmes.

Le domaine de réseau a été abordé par chacun des membres de l'équipe durant le semestre passé, nous sommes donc tous au même niveau de compétence. En ce qui concerne l'automatique et la transmission, les compétences de chacun varient en fonction de leurs formations précédentes.

2.2. Compétences en programmation

	Java	C	Python	Arduino	Android	Web
Léo	3	1	3	1	1	1
François	4	1	1	1	1	1
Max	5	3	2	3	1	1

Tableau 2 : Compétences en langages de programmation

Grâce à ce deuxième tableau (Tableau 2), on se rend compte que dans le domaine de java, comme pour l'électronique, Max est le plus compétent. Pour ce qui est des autres domaines de programmation (Android, C, Arduino...) le niveau de compétences global est assez faible.

3. Affectation des tâches et structure de l'équipe

D'après les tableaux, on peut conclure que l'électronique et les réseaux sont les points forts tout comme le langage de programmation Java. En revanche la transmission et les autres langages de programmation sont considérés comme des points faibles.

Comme vu plus haut, les tableaux ont pour but de récapituler toutes les compétences, que ça soit en termes de compétences globales ou en termes de langages de programmation. Après avoir étudié ces tableaux on en a conclu la répartition suivante :

François : Réseau

Max : Transmission

Léo : Électronique

Pour ce qui est du développement, tous les membres de l'équipe vont participer car il s'agit de la partie prépondérante de notre projet. Cependant nous avons défini des sous parties auxquelles nous avons attribué un responsable :

- **Partie application** : François
- **Partie serveur** : Max
- **Partie traitement du langage** : Léo

Même si nous avons fait ces choix, ils ne sont pas fixes. C'est à dire que si un des membres de l'équipe a besoin d'aide dans un des domaines, l'un des deux autres membres pourra l'aider. Pour chaque étape du projet nous allons nous répartir les tâches à effectuer en fonction de nos capacités.

4. Présentation de l'objectif

Les objectifs de ce projet sont multiples. Le principal est la création d'un boîtier intelligent qui permettra de contrôler des équipements domotiques dans une maison. Ce boîtier devra communiquer avec ces équipements en non-filaire. Comme équipements, nous voulons pouvoir contrôler n'importe quels types d'équipements domotiques, que ce soit des lampes, des volets, des thermostats etc.

L'interaction entre l'utilisateur et le système se fera à travers une chat box textuelle à l'intérieur d'une application. Cette application devra être multiplateformes et accessible par tous les membres d'une famille, utilisant des comptes différents.

Nous voulons une solution domotique intuitive et facile d'utilisation. Elle doit être pratique et on ne veut pas que les données soient stockées sur des serveurs externes à notre système.

Nous avons défini dans le cahier des charges nos objectifs et les fonctionnalités que nous voulions implémenter en fonction de leur priorité. Mais nous avons également un objectif secondaire qui est de créer un site web permettant de suivre l'avancement du projet.

5. Introduction au cahier des charges

La fonction principale de notre boîtier est de contrôler les différents équipements qui sont connectés dans la maison. L'utilisateur devra être en mesure, à partir d'une application et via une chatbox textuelle, de contrôler l'ensemble des objets connectés en local (pas d'exportations de données sur internet). Notre système sera composé de plusieurs parties. La partie boîtier, la partie application, la partie serveur, traitement du langage et équipements domotiques. Nous avons répertorié les spécificités de ces parties dans différents tableaux. Nous avons défini les fonctionnalités que nous voulions implémenter selon deux catégories : FP (Fonctionnalité Principale) et FC (Fonctionnalité Complémentaire).

- **Partie Boîtier** : Notre boîtier contiendra la partie serveur et la partie traitement du langage qui seront hébergés sur des cartes Raspberry Pi. Il faut qu'il soit assez grand pour accueillir un nombre de cartes suffisant et qu'il soit possible de l'agrandir ou d'accéder facilement aux cartes à l'intérieur.

Partie Boîtier		
Fonctionnalités	Contraintes	Solutions envisagées
FP1 : Contenir les cartes Raspberry Pi	La taille de la base doit être suffisante pour accueillir les cartes + les câbles tout en étant assez aéré.	Taille : 12x10x25cm de forme rectangulaire Trou d'aération prévu pour la ventilation des Raspberry Pi
FC1 : Être esthétique et discret	Le boîtier est placé dans une maison donc il doit se fondre dans la maison	Design simple, de couleur discrète, réalisé par imprimante 3D

FC2 : Permettre l'accueil de nouveaux Raspberry Pis et laisser la possibilité d'ajouter des cartes pour gérer les différents protocoles réseaux	Ne doit pas être trop grand et rester stable. Possibilité d'accéder facilement à toutes les cartes. La taille doit être adaptée aux nombres de cartes à l'intérieur.	Boîtier comportant des parties montées comme des "Legos". Une base avec un rac et un espace pour les câbles avec plusieurs modules additionnels que l'on empile avec un couvercle.
--	---	--

- **Partie Application** : Notre application devra être adaptée aux besoins de l'utilisateur. Nous voulons implémenter différentes fonctionnalités dont certaines plus importantes et prioritaires. Elle devra être connectée à la partie serveur afin de transmettre les demandes de l'utilisateur.

Partie Application		
Fonctionnalités	Contraintes	Solutions envisagées
FP1 : Être capable d'interagir directement avec chaque équipement	Identifier spécifiquement chaque équipement domotique	Associer des mots clefs à chaque équipement dans une BDD nous permettant de les identifier.
FP2 : Communiquer avec le serveur depuis l'intérieur de la maison	Choix d'un protocole de communication	On peut utiliser soit Bluetooth, soit le Wifi. Pour ce qui est de la distance de communication nécessaires les deux répondent à notre fonction.
FP3 : Envoyer des commandes via messages textes	Où traiter ces commandes Où taper les commandes	Utiliser un système de neurones capable de catégoriser les différentes demandes. Les commandes seront tapées dans une chatbox sur l'application textuellement.
FP4 : Création d'un compte utilisateur	Le stockage doit être géré	Système d'enregistrement et de stockage
FP5 : Pouvoir créer des scénarios personnalisés	Difficultés à enregistrer les scénarios et à programmer. Où créer des scénarios ?	Scénarios propres à chaque utilisateur. L'utilisateur pourra programmer ses propres scènes pour ensuite avoir la possibilité de la lancer par commande textuelle ou automatiquement en fonction de la date.

		Les scénarios pourront être programmés dans l'interface configuration, pour ce faire nous envisageons la possibilité d'utiliser des DSL qui sont des langages dédiés déjà existant.
FP6 : Développer l'application sur la plateforme Android		Développer cette application à l'aide d'Android Studio. À la vue de nos compétences, c'est ce qui nous semble le plus réalisable.
FC1 : Être simple d'utilisation et épurée		L'interface doit être simple, intuitive, afin que tout le monde puisse l'utiliser
FC2 : Commander les équipements depuis l'extérieur de la maison	Difficulté à communiquer avec la partie serveur	Utilisation d'internet pour interagir
FC3 : Visualiser l'état des équipements domotiques sur l'application		Recevoir les retours d'états des équipements le permettant
FC4 : Développer un site Web		Utiliser les connaissances du module WO
FC5 : Développer l'application en multi plateformes	Être capable de s'adapter	Développer l'application sous différents OS

- **La partie serveur :** Dans notre boîtier, il y aura une partie serveur géré via un ou plusieurs Raspberrys pis. Ces derniers devront recevoir les demandes des utilisateurs et les faire analyser par la partie traitement du langage. La partie traitement de langage devra répondre au serveur en renvoyant la demande programme. Le serveur envoie ensuite la ou les commandes appropriées aux réseaux d'objets connectés. Le système doit être capable de communiquer avec les objets connectés en Zigbee et/ou avec les protocoles en 433 et 868 Mhz car les équipements que nous aurons ne pourrons communiquer qu'avec ces protocoles. Cela couvre de nombreux protocoles (voir état de l'art). Chaque membre d'une famille devra créer son propre compte qui contrôlera les objets de la maison et il ne sera pas possible de connecter plusieurs personnes au même compte. Il y aura potentiellement des soucis si plusieurs personnes veulent allumer ou éteindre une lampe, et nous allons donc devoir définir des droits utilisateurs pour contrer ce problème.

Partie Serveur		
Fonctionnalité	Contraintes	Solutions envisagés
FP1 : Relayer les demandes utilisateur à la partie traitement du langage.	Communication entre la partie serveur et traitement du langage	Nous pourrons faire communiquer ces deux parties en filaire puisqu'elles seront situées au même endroit.
FP2 : Gérer les comptes utilisateurs	Difficultés d'enregistrement et de stockage	Création d'une base de données permettant d'enregistrer les données importantes
FP3 : Assurer une sécurité de l'information	Les personnes extérieures ne doivent pas avoir la possibilité de contrôler nos équipements domotiques	Mot de passe pour le compte utilisateur, suffisant pour le moment mais il faudra penser à augmenter la sécurité dans un second temps.
FP4 : Envoyer les ordres aux équipements domotiques reçus par la partie traitement du langage		Utilisation du protocole utilisé par l'équipement en question. Ajout de modules protocolaires sur les ports USB des cartes.
FP5 : Communiquer en Zigbee dans un premier temps (avant de communiquer avec tous types de protocoles)		Brancher un module protocolaire Zigbee sur notre serveur et faire communiquer ce dernier avec des équipements utilisant ce protocole.
FC1 : Analyser les informations des capteurs et retour d'états des équipements domotiques	Avoir assez de puissance pour analyser ces données	Augmenter le nombre de cartes dans la partie serveur
FC2 : Communiquer avec tous types de protocoles 433 et 868 Mhz + Zigbee	Intégration des modules protocolaires à notre serveur + s'assurer du bon fonctionnement	Faire des tests avec chaque nouveau protocole afin d'adapter notre système

- **La partie traitement du langage** : Pour cette partie, il faudra prévoir un ou plusieurs Raspberry qui recevront les demandes utilisateurs via le serveur et les traiteront afin de renvoyer la demande programme au serveur. Le fonctionnement de cette partie sera basé sur un réseau de neurones (voir état de l'art). Ce réseau de neurones devra nous permettre de réaliser les fonctionnalités décrites ci-dessous.

Partie traitement du langage		
Fonctionnalité	Contraintes	Solutions envisagés
FP1 : Comprendre un message texte contenant seulement des mots clefs	Reconnaître les mots clefs	Reconnaître certains mots envoyés par l'utilisateur et y associer des commandes.
FC1 : Comprendre des phrases sans fautes	Récupérer les informations importantes d'une phrase	Utilisation du Deeplearning pour avoir des phrases modèles
FC2 : Comprendre les phrases avec fautes	Difficultés de faire correspondre un mot erroné à un mot correct	Utilisation du Deeplearning afin de faire le lien avec la base de données et se rapprocher du modèle
FC3 : Réponse aux messages (retour d'état + formule de politesse)		Intégrer un retour d'état par message suite à l'exécution d'une commande

Exemple de traitement du langage et de reconnaissance avec Core NLP :

Named Entity Recognition:

1	<div> <div>Person</div> <div>Loc</div> <div>ORDINAL</div> <div>Location</div> </div> President Xi Jinping of China, on his first state visit to the United States, showed off his familiarity with <div> <div>Misc</div> <div>Date</div> <div>Time</div> </div> American history and pop culture on Tuesday night.
---	--

- **Les objets connectés** : Le boîtier contrôlera des équipements. Il faudra donc, avant tout, que la maison soit équipée en objets connectés. Ils devront être capable de communiquer avec la partie serveur en indiquant tous types de données à propos d'eux-mêmes. Ils devront donc pouvoir communiquer en Zigbee et/ou avec les protocoles 433 et 868Mhz (la raison est expliquée dans la partie serveur). Le boîtier devra être capable de commander tous types d'objets connectés tels que des lampes, des volets... Il n'y a pas de fonctionnalités spéciales à implémenter dans cette partie. Il faudra cependant faire attention à la compatibilité des protocoles de transmissions entre notre partie serveur et les objets connectés.

En conclusion, nous avons défini certaines fonctionnalités prioritaires que nous voulons implémenter, et d'autres secondaires. Nous voulons que notre système soit flexible, c'est à dire que l'utilisateur puisse ajouter ou enlever des équipements domotiques de sa maison de manière facile et intuitive.

En ce qui concerne les délais, l'équipe devra rendre plusieurs livrables afin de rendre compte de l'avancement du projet. De plus des soutenances sont prévues pour présenter le projet oralement. Enfin, ce projet de boîtier intelligent doit être terminé avant le 25 mai 2018.

6. Mode de fonctionnement

6.1. Au sein de l'équipe

Afin de réussir ce projet, il est important de bien communiquer dans le groupe pour être le plus productif possible. Chaque membre du groupe va faire des recherches de son côté tout au long de la semaine et tous les vendredis, il y aura une mise en commun de toutes les informations trouvées durant la semaine. Cela permettra de régler directement les problèmes survenus au cours de la semaine. Afin de communiquer, nous avons mis en place un Google Drive et nous discutons via Messenger. Pour se répartir les tâches nous utilisons Trello, cela nous permet également de planifier tout ce que l'on a à faire mais aussi de valider les objectifs qui sont terminés.

6.2. Tâches à accomplir

Il y aura différentes formes de tâches à accomplir. En effet dans un premier temps, les tâches seront basées sur beaucoup de recherches et de documentations sur les produits existants et sur tous les protocoles qu'ils utilisent. Ensuite il va falloir que l'on détermine quels équipements nous allons utiliser dans notre maison et quels matériels nous allons avoir besoin pour mettre en œuvre notre boîtier connecté. Enfin la dernière tâche à accomplir sera la soutenance qu'il faudra préparer.

	Pla...	Nom de tâche	Durée	Début	Fin
1	★	Projet Boîtier Connecté	45 jours	03/10/2017	04/12/2017
2	★	Soutenance n°1	6 jours	01/12/2017	08/12/2017
3	★	Journée Porte Ouverte	6 jours	26/01/2018	03/02/2018
4	★	Partie Recherche et Formation	34 jours	03/10/2017	17/11/2017
5	★	État de l'art	9 jours	03/10/2017	13/10/2017
6	★	Recherches sur les différents protocoles de transmission	9 jours	03/10/2017	13/10/2017
7	★	Formation sur les fonctionnalités de base du Raspberry pi	9 jours	03/10/2017	13/10/2017
8	★	Détermination du cahier des charges	9 jours	03/10/2017	13/10/2017
9	★	Affectation des tâches	9 jours	03/10/2017	13/10/2017
10	★	Détermination des équipements nécessaires	26 jours	13/10/2017	17/11/2017
11	★	Partie Communication	161 jours	13/10/2017	25/05/2018
12	★	Délivrable n°1	9 jours	03/10/2017	13/10/2017
13	★	Rassembler les informations	9 jours	03/10/2017	13/10/2017
14	★	Mise en page + webographie / bibliographie	9 jours	03/10/2017	13/10/2017
15	★	Délivrable n°2	26 jours	13/10/2017	17/11/2017
16	★	Délivrable n°3	16 jours	17/11/2017	08/12/2017
17	★	Mise en ligne d'un site	136 jours	17/11/2017	25/05/2018
18	★	Création du site	136 jours	17/11/2017	25/05/2018
19	★	Skin du site	136 jours	17/11/2017	25/05/2018
20	★	Mise à jour de l'avancement du projet	136 jours	17/11/2017	25/05/2018
21	★	Partie boîtier + Chat Box	156 jours	20/10/2017	25/05/2018
22	★	Développement d'un système basique	46 jours	20/10/2017	22/12/2017
23	★	Interprétation de commandes rudimentaires par la chat box	46 jours	20/10/2017	22/12/2017
24	★	Intégration d'un objet connecté avec le Raspberry pi	46 jours	20/10/2017	22/12/2017
25	★	Évolution vers un système plus complexe	55 jours	08/01/2018	23/03/2018
26	★	Interprétation de commande plus complexes (Base de données par exemple)	55 jours	08/01/2018	23/03/2018
27	★	Établir la connexion entre Raspberry pi et tous types d'objets connectés	55 jours	08/01/2018	23/03/2018
28	★	Optimisation du système	46 jours	23/03/2018	25/05/2018
29	★	Prise en compte d'erreurs dans les demandes de l'utilisateur	46 jours	23/03/2018	25/05/2018
30	★	Réalisation de fonctions plus complexes	46 jours	23/03/2018	25/05/2018
31	★	Création du boîtier	100 jours	08/01/2018	25/05/2018

Tableau 3: Récapitulation du diagramme de Gantt

Comme vous pouvez le voir sur le tableau ci-dessus (tableau 3), nous avons divisé notre projet en trois grandes parties plus les dates pour les soutenances. La première partie correspond à la phase d'introduction au projet : les recherches et l'apprentissage des notions de bases pour pouvoir comprendre et se familiariser avec l'environnement du Raspberry, les différents protocoles de transmission, etc. La seconde partie correspond à la communication du projet : livrables et le développement du site web. Nous ferons évoluer ce site au fur et à mesure de notre projet. Enfin, la dernière partie concerne le développement même du boîtier connecté et de la chatbox. Nous avons décidé de travailler en méthode agile pour cette partie. En effet, nous allons développer notre système en partant de fonctionnalités de bases, puis nous allons le faire évoluer petit à petit en implémentant des fonctionnalités de plus en plus complexes. Cela nous permettra d'avoir un système opérationnel à n'importe quel moment dans l'avancement de notre projet.

Le diagramme de Gantt se trouve en annexe.

7. Identification des risques

Problème	Solution à mettre en œuvre
Mise à niveau dans les différents langages de programmation (Java, C, Python)	Essayer d'utiliser au maximum nos compétences acquises lors de nos différents cursus et se documenter un maximum afin de rattraper les lacunes.
Connaissance des protocoles de transmission (Radio, Zigbee, Zwave, EnOcean)	Procéder à une recherche et un état de l'art approfondis.
Gestion du temps et des ressources	Mise en place d'outils permettant une bonne gestion et répartition des tâches pour ainsi ne pas perdre de temps. Avoir une forte communication au sein du groupe pour éviter toutes confusions.
Gestion des protocoles réseaux	Se former afin de comprendre comment ils fonctionnent.
Connaissance du Raspberry pi	Se former pour être capable de l'utiliser (connaître les possibilités de la carte, quels modules sont compatibles, les limites de son utilisation)
Matériel non livré	Même sans matériel, il nous sera possible de réaliser la partie application de notre projet ainsi que la partie traitement du langage. Nous pourrions donc tout de même créer une chat box textuelle capable de traiter l'information envoyée

Tableau 4 : Récapitulation des risques lors de notre projet

8. État de l'art

Qu'est-ce que la maison connectée ?

Dans une maison connectée on peut contrôler ses équipements à distance et avoir des informations sur leur état en permanence. La sécurité de la maison est renforcée par des alarmes et des programmes automatiques nous permettent de faire des économies d'énergie. Tous ces équipements sont connectés par des protocoles de transmission différents. Pour que tous ces objets communiquent entre eux, il faut qu'un commutateur intelligent récolte toutes ces données et les analyse, et également comprendre des demandes provenant de l'utilisateur. De nombreuses utilisations sont alors possibles.

8.1. Différentes Box domotiques

8.1.1. Jeedom Smart

C'est la box domotique spécialisée pour Jeedom (logiciel de domotique présenté précédemment). Sa taille est légèrement plus grande qu'un Raspberry Pi. Elle fonctionne donc avec Jeedom et elle est proposée en 4 différentes versions : Z-wave, EnOcean, avec ou sans

RFXcom. Elle possède plusieurs connectiques (ports USB, HDMI, Ethernet) qui permettent d'ajouter de nouveaux protocoles si besoin. [27]

La box est basée sur un Odroid C2 avec un processeur 1.5GHz et 2Go de RAM.

Une version pro est disponible, plus puissante, et compatible avec plus de protocoles.

Caractéristiques	Jeedom Smart	ZenBox
Dimensions (l x l x h)	21.5 x 14.5 x 8.5 cm	Variable mais min 12 x 10 x 25cm
Protocoles compatibles	Nativement soit Z-wave ou EnOcean	Zigbee et protocoles en 433 et 868 Mhz
Connectiques	4 ports USB, HDMI et Ethernet	Plusieurs ports USB en fonction du nombre de Raspberry Pi, ports HDMI et Ethernet.
Logiciel	Logiciel openSource Jeedom multiplateformes	Logiciel application Android et site web Commandes textuelles supportées à travers l'application
Carte	Odroid C2 RAM : 2Go Cadence Processeur: 1.5GHz Stockage : DD Emmc 8 Go WIFI : non Bluetooth : non	Raspberry Pi V2/V3 RAM: 1Go/1Go Cadence Processeur : 900Mhz/1.2 Ghz Stockage : MicroSD WIFI : non/oui Bluetooth : non/oui
Accès à distance	Oui	Oui

Tableau 5 : comparaison entre box Jeedom Smart et la ZenBox

En conclusion on peut voir que la box Jeedom Smart semble plus puissante et plus stable que notre box. En effet la solution Jeedom se base sur une carte Odroid C2 avec stockage Emmc, qui est bien plus fiable qu'une carte microSD. En revanche, notre solution est compatible avec bien plus de protocoles, et nos cartes V3 contiennent déjà de quoi les connecter en Wifi ou Bluetooth. Nous pouvons donc couvrir plus de protocoles de transmission, et même si notre box est plus haute, elle ne prend pas trop de place au point de gêner l'utilisateur. Nous pouvons commander nos équipements domotiques par commande textuelle, contrairement aux utilisateurs de la box Jeedom Smart.



Figure 1 : Jeedom Smart

8.1.2. Eedomus+

C'est une box domotique qui peut se connecter à de nombreux équipements domotiques y compris les systèmes de reconnaissance vocaux echo, home etc. Elle est compatible avec plusieurs protocoles de transmission, notamment Z-wave qui est compatible nativement, et EnOcean et les protocoles en 433Mhz qui sont optionnels. On peut trouver une liste de périphériques auxquels elle peut se connecter sur le site internet de la box.

Elle s'appuie sur une application mobile ou sur ordinateur à partir d'un site web. Il est possible de télécharger un API Eedomus+ permettant d'aller plus loin dans les configurations. [28]

Caractéristiques	Eedomus+	ZenBox
Dimensions (l x l x h)	14.7 x 14.7 x 3.5 cm	Variable mais min 12 x 10 x 25cm
Protocoles compatibles	Z-wave intégré EnOcean (optionnel) 433Mhz (optionnel)	Zigbee et protocoles en 433 et 868 Mhz
Connectiques	1 port Ethernet 2 ports USB 4 ports d'extension filaire Haut-parleur intégré	Plusieurs ports USB en fonction du nombre de Raspberry Pi, ports HDMI et Ethernet.
Logiciel	Eedomus avec un site web ou application multiplateforme (Apple, Android, MAC, PC, Windows8) Fonctionne avec les assistants vocaux (Alexa, Google Assistant). Logiciel propriétaire.	Logiciel application Android et site web Commandes textuelles supportées à travers l'application

Carte	Eedomus RAM : 512 Mo Cadence Processeur : 1Ghz Stockage : 4Go eMMC WIFI :	Raspberry Pi V2/V3 RAM : 1Go/1Go Cadence Processeur : 900Mhz/1.2 Ghz Stockage : MicroSD WIFI : non/oui Bluetooth : non/oui
Service cloud	Oui. Permet : Accès à distance Sauvegarde des données Notifications (email,SMS,push)	Non
Accès à distance	Oui	Oui

Tableau 6 : Comparaison entre Eedomus et la ZenBox

On peut voir que la Box eedomus permet une gestion à distance de sa domotique mais cela doit passer par le cloud. Contrairement à notre box qui permet une gestion à distance sans stocker les données extérieurement à la maison. Le système eedomus est compatible avec les assistants vocaux et tout est contrôlable à travers l'application tandis que notre solution propose un contrôle via chatbox textuelle. Les protocoles de transmission compatibles sont plus nombreux avec notre box.



Figure 2 : Eedomus+

8.1.3. Home center 2

Ce système de gestion de bâtiments est disponible à partir d'une application mobile ou d'un site web. Il est compatible avec les produits Z-wave en particulier. Une bibliothèque de plugins est déjà installée dans la box et de nombreux équipements domotiques sont disponibles avec cette box également.

Son logiciel FIBARO propriétaire permet de programmer en LUA des scripts pour apporter de nouvelles fonctionnalités. Il est possible de programmer des scènes et il peut utiliser la fonction géolocalisation pour en déclencher. [29] & [30]

Caractéristiques	Home center 2	ZenBox
Dimensions (l x l x h)	22.5 x 18.5 x 4.2 cm	Variable mais min 12 x 10 x 25cm
Protocoles compatibles	Z-wave et particulièrement les équipements Z-wave de Fibaro	Zigbee et protocoles en 433 et 868 Mhz
Connectiques	1 Port Ethernet 2 ports USB	Plusieurs ports USB en fonction du nombre de Raspberry Pi, ports HDMI et Ethernet.
Logiciel	Logiciel Fibaro multiplateformes et site web Commandes vocales supportées avec Amazon Alexa et Siri	Logiciel application Android et site web Commandes textuelles supportées à travers l'application
Carte	RAM : 1Go + 2Go pour le système Fibaro Cadence Processeur : 1.6 Ghz Stockage : 4Gb SSD pour les sauvegardes et un autre pour le système	Raspberry Pi V2/V3 RAM : 1Go/1Go Cadence Processeur : 900Mhz/1.2 Ghz Stockage : MicroSD WIFI : non/oui Bluetooth : non/oui
Service cloud	Non	Non
Accès à distance	Oui	Oui

Tableau 7 : Comparaison entre Home center 2 et la ZenBox

On peut voir que la Box Home Center 2 utilise un logiciel propriétaire et est bien plus puissante que la nôtre. En effet, c'est la plus puissante sur le marché, et elle stocke ses données en interne. Cependant, elle n'est compatible qu'avec les équipements Z-wave, tandis que la nôtre est compatible avec n'importe quel protocole. La commande vocale est supportée pour la box Home center 2 et des notifications push et SMS sont possibles avec un accès à distance de la configuration domotique.



Figure 3 : Home Center 2

8.1.4. Somfy Box

Cette box est appelée TaHoma et est compatible avec les technologies io-homecontrol et RTS. Il est possible de commander les équipements domotiques 1 à 1 ou créer des scénarios. Elle utilise l'interface du même nom, TaHoma, qui permet de piloter les équipements domotiques. Elle est disponible sur iOS et Android, sur PC, et tablettes également. Il est possible de la rendre compatible avec d'autres protocoles via adaptateur USB. [31]

Caractéristiques	Somfy Box	ZenBox
Dimensions (l x l x h)	11 x 7.1 x 2.8 cm	Variable mais min 12 x 10 x 25cm
Protocoles compatibles	io-homecontrol et RTS nativement Z-Wave et EnOcean par adaptateur USB	Zigbee et protocoles en 433 et 868 Mhz
Connectiques	1 port Ethernet 1 micro USB 1 port USB	Plusieurs ports USB en fonction du nombre de Raspberry Pi, ports HDMI et Ethernet.
Logiciel	TaHoma logiciel propriétaire multiplateforme. Compatible avec amazon Alexa. Possibilité de créer des scénarios	Logiciel application Android et site web Commandes textuelles supportées à travers l'application

Carte		Raspberry Pi V2/V3 RAM : 1Go/1Go Cadence Processeur : 900Mhz/1.2 Ghz Stockage : MicroSD WIFI : non/oui Bluetooth : non/oui
Service cloud	En fonction des équipements domotiques utilisés	Non
Accès à distance	Oui	Oui

Tableau 8 : Comparaison entre Somfy Box et la ZenBox

La box de Somfy n'est pas compatible avec tous les protocoles et donc pas avec tous les équipements, contrairement à notre box. Certaines données sont stockées sur le Cloud en fonction des équipements utilisés. D'autres données sont stockées en interne dans la box. De nombreuses autres solutions clef en main sont disponibles mais ne seront pas analysées ici.



Figure 4 : Somfy box

8.2. Outils d'assistances vocales

On a pu voir que plusieurs boxs de domotiques utilisaient des services d'assistance vocales, notamment Alexa d'Amazon ou Google. Ce ne sont pas des systèmes domotiques complets mais ils permettent de contrôler certains équipements domotiques et d'analyser des commandes vocales de l'utilisateur et d'y répondre. Ils peuvent également être intégrés dans des systèmes domotiques plus complexes.

8.2.1. Amazon Echo

Amazon Echo est un boîtier connecté développé par le lab126 d'Amazon, disponible uniquement en anglais et en allemand actuellement. Il est disponible depuis le 23 juin 2015 aux Etats-Unis et n'est pas encore disponible en France. [10] & [11] & [12]

Le boîtier en lui-même n'est constitué que d'enceintes permettant de diffuser le son, de microphones et d'un micro-ordinateur. Il est connecté au système d'Amazon permettant de traiter les demandes des utilisateurs. [32]

Un micro-ordinateur à l'intérieur du boîtier est assez intelligent pour effectuer un certain nombre de tâches. Et lorsqu'il détecte les mots-clés permettant d'activer la reconnaissance vocale et vos demandes commencent à être enregistrées. Elles sont ensuite redirigées vers le service cloud d'Amazon qui les analyse. Le service qui analyse les commandes s'appelle Alexa Voice Services (AVS). Il transpose les commandes vocales en commandes textuelles interprétables par les équipements. Lorsque l'on effectue une demande, comme allumer une lumière, Alexa va envoyer une demande à Echo qui va la transférer à la lumière afin de l'allumer.

Les technologies Google ou Apple fonctionnent sur le même principe, les données sont simplement traitées par d'autres serveurs.

Des questions de sécurité de données se posent. Les stations de reconnaissance vocales semblent écouter en permanence et on ne peut être certains de comment et où sont stockées les données enregistrées.



Figure 5 : Amazon Echo & Alexa

8.2.2. Google Home

C'est un boîtier connecté avec système de reconnaissance vocale développé par Google, concurrent d'Amazon Echo. Il est annoncé en mai 2016 et disponible en octobre 2016. Il devient disponible sur le marché français en août 2017. Il devance donc Amazon ou Apple en France. Il fonctionne de la même manière que ses concurrents. [13]



Figure 6 : Google Home

8.2.3. HomePod

C'est une enceinte intelligente présentée par Apple le 5 juin 2017. Elle sera disponible en décembre 2017 seulement aux Etats-Unis. C'est un produit similaire et concurrent à Amazon Echo et Google Home. Il utilise Siri pour la reconnaissance vocale et peut se connecter à tout objet connecté d'Apple. Le prix annoncé est plus élevé que Google home ou Echo. Ce dernier est justifié par une qualité supérieure à celle de ses concurrents. [14]



Figure 7 : Homepod

8.3. Différents logiciels domotiques

8.3.1. Jeedom Interaction

Jeedom est un système d'interaction. Ce système peut nous être très utile car il permet de réaliser des actions à l'aide de commande texte. Ces actions sont envoyées à Jeedom à l'aide d'autres applications (Sarah, Domo Widget, VocalDom ou des sms).

Exemple, pour éteindre une lumière je peux être amené à dire :

- Éteint la lumière dans le couloir.
- Éteindre la lampe du couloir.
- Arrêter la lumière dans le couloir.
- Éteindre la lampe dans le couloir.
- Etc...

Le logiciel Jeedom est Open Source, il permet un accès total au logiciel qui gère la domotique. Jeedom est compatible avec différents protocoles comme le Z-Wave, le RFXcom, le RTS SOMFY, le EnOcean, le xPL, etc... Le système de plugins, via le Market Jeedom, permet de garantir une compatibilité avec de nombreux protocoles actuels et futurs.

Chaque utilisateur peut créer sa propre domotique Jeedom. À l'aide des widgets, des vues et des designs, vous avez une totale liberté pour imaginer votre propre interface si vous le souhaitez. [7]

	Jeedom	ZenBox
Multiplateforme	Android	Application Android et serveur web
Distance	Seulement dans la maison	Dans la maison mais on veut pouvoir par la suite communiquer depuis l'extérieur

Plusieurs comptes	Utilisateur créer sa propre domotique	Chaque utilisateur est différent et doit pouvoir se créer un compte pour gérer ses équipements
OpenSource / Propriétaire	OpenSource	On veut un logiciel OpenSource
Scénario	Oui	L'utilisateur doit pouvoir créer ses propres scénarios
Retour d'états possible	Oui	L'utilisateur doit avoir un retour d'état lorsqu'il commande un équipement
Commande voix / texte	Texte et voix grâce à vocaldom	Transmettre les commandes via une commande texte

Tableau 9 : Comparaison entre Jeedom et la ZenBox

8.3.2. Domoticz

Le système Domoticz est un autre système de solution de domotique permettant de contrôler les équipements domotiques dans notre maison. C'est un logiciel open Source multiplateformes, notamment disponible sur les plateformes embarquées et Raspberry Pi. L'interface utilisateur est une webApp compatible avec tous les navigateurs. Le système peut utiliser de nombreux protocoles de transmission et fonctionne avec de nombreux composants. On peut trouver une liste de ces équipements compatibles dans le manuel disponible sur leur site internet.

Le système offre la possibilité de partager des équipements avec d'autres personnes. Une personne a le rôle d'administrateur et peut partager les données récoltées par ses capteurs. De nombreux graphiques permettant d'analyser les données sont disponibles.

Le logiciel est particulièrement bien compatible avec les cartes Raspberry Pi. [22]

	Domoticz	ZenBox
Multiplateforme	Oui mais seulement en serveur web	Application Android et serveur web
Distance	Seulement dans la maison	Dans la maison mais on veut pouvoir par la suite communiquer depuis l'extérieur
Plusieurs comptes	Partage d'équipements	Chaque utilisateur est différent et doit pouvoir se créer un compte pour gérer ses équipements
OpenSource / Propriétaire	OpenSource	On veut un logiciel OpenSource

Scénario	Oui	L'utilisateur doit pouvoir créer ses propres scénarios
Retour d'états possible	Oui	L'utilisateur doit avoir un retour d'état lorsqu'il commande un équipement
Commande voix / texte	Texte via ligne de commande	Transmettre les commandes via une commande texte

Tableau 10 : Comparaison entre Domoticz et la ZenBox

8.3.3. Yadoms

Yadoms est une autre solution domotique « made in France » open source et multiplateforme. Disponible également sur Raspberry Pi. Il est possible de construire des scénarios contrôlant plusieurs équipements connectés pour automatiser des tâches. Possibilité de recevoir des SMS d'alertes. Des graphiques sont disponibles pour analyser les données collectées. [23]

De nombreux appareils sont supportés par le logiciel et des mises à jour régulières sont effectuées. On peut trouver une liste de tous ces appareils sur leur site internet

	Yadoms	ZenBox
Multiplateforme	Oui mais seulement via serveur web	Application Android et serveur web
Distance	Seulement dans la maison	Dans la maison mais on veut pouvoir par la suite communiquer depuis l'extérieur
Plusieurs comptes		Chaque utilisateur est différent et doit pouvoir se créer un compte pour gérer ses équipements
OpenSource / Propriétaire	OpenSource	On veut un logiciel OpenSource
Scénario	Oui	L'utilisateur doit pouvoir créer ses propres scénarios
Retour d'états possible	Oui	L'utilisateur doit avoir un retour d'état lorsqu'il commande un équipement
Commande voix / texte	Via le logiciel	Transmettre les commandes via une commande texte

Tableau 11 : Comparaison entre Yadoms la ZenBox

8.3.4. Calaos

Cet autre logiciel de domotique est apparu en 2013. Le code est open source. Ce logiciel n'est pas compatible avec le protocole de transmission Z-Wave. Il est également possible de créer des scénarios avec ce logiciel.

La couche logicielle est composée d'un serveur, d'une interface, d'une application web, d'un outil de configuration, d'applications mobiles et d'un système complet pour OS.

Il ne supporte que les Premoboards, Cubieboard, Raspberry Pi, et les machines Intel 32 et 64 bits.

Il permet de piloter les automates de marque WAGO en particulier, GPIO d'une carte Raspberry par exemple, Zibase, one Wire, et d'autres. [24]

	Calaos	ZenBox
Multiplateforme	Oui + serveur web	Application Android et serveur web
Distance	Seulement dans la maison	Dans la maison mais on veut pouvoir par la suite communiquer depuis l'extérieur
Plusieurs comptes	Gérable depuis la connexion de la maison	Chaque utilisateur est différent et doit pouvoir se créer un compte pour gérer ses équipements
OpenSource / Propriétaire	OpenSource	On veut un logiciel OpenSource
Scénario	Oui	L'utilisateur doit pouvoir créer ses propres scénarios
Retour d'états possible	Oui	L'utilisateur doit avoir un retour d'état lorsqu'il commande un équipement
Commande voix / texte	Via l'application ou la Webapp, appuyer sur ON/OFF ou modifier les valeurs	Transmettre les commandes via une commande texte

Tableau 12 : Comparaison entre Calaos et la ZenBox

8.3.5. Home Assistant

C'est un logiciel développé entièrement en Python multiplateformes. Elle est d'abord destinée au Raspberry Pi. C'est un logiciel open source également. Une liste des équipements supportés est disponible sur leur site internet. [25]

	Home Assistant	ZenBox
Multiplateforme	Oui(MAC/OS/Ubuntu)	Application Android et serveur web

Distance	Seulement dans la maison	Dans la maison mais on veut pouvoir par la suite communiquer depuis l'extérieur
Plusieurs comptes		Chaque utilisateur est différent et doit pouvoir se créer un compte pour gérer ses équipements
OpenSource / Propriétaire	OpenSource	On veut un logiciel OpenSource
Scénario	Oui	L'utilisateur doit pouvoir créer ses propres scénarios
Retour d'états possible	Oui	L'utilisateur doit avoir un retour d'état lorsqu'il commande un équipement
Commande voix / texte	Texte via ligne de commande	Transmettre les commandes via une commande texte

Tableau 13 : Comparaison Home Assistant et la ZenBox

8.3.6. OpenHab

Cet autre logiciel de solution domotique est un projet open source qui supporte de nombreux protocoles de transmission et d'équipements domotiques (200+). Il est écrit complètement dans le langage Java et est basé sur Eclipse SmartHome. Il utilise Apache Karaf avec Eclipse Equinox et Jetty comme serveur web. [26]

	OpenHab	ZenBox
Multiplateforme	Oui + serveur web	Application Android et serveur web
Distance	Seulement dans la maison	Dans la maison mais on veut pouvoir par la suite communiquer depuis l'extérieur
Plusieurs comptes		Chaque utilisateur est différent et doit pouvoir se créer un compte pour gérer ses équipements
OpenSource / Propriétaire	OpenSource	On veut un logiciel OpenSource
Scénario	Oui	L'utilisateur doit pouvoir créer ses propres scénarios
Retour d'états possible	Oui	L'utilisateur doit avoir un retour d'état lorsqu'il commande un équipement
Commande voix / texte	Commande à programmer via Eclipse	Transmettre les commandes via une commande texte

Tableau 14 : Comparaison entre OpenHab et la ZenBox

8.3.7. TaHoma

Ce logiciel est le logiciel propriétaire vendu avec la box du même nom par SOMFY. Il offre une interface intuitive accessible depuis un ordinateur, tablette ou smartphone.

	TaHoma	ZenBox
Multiplateforme	Oui. Disponible sur ordinateur, smartphone et tablette.	Application Android et serveur web
Distance	Pilotage à domicile et à distance	Dans la maison mais on veut pouvoir par la suite communiquer depuis l'extérieur
Plusieurs comptes	Oui	Chaque utilisateur est différent et doit pouvoir se créer un compte pour gérer ses équipements
OpenSource / Propriétaire	Propriétaire	On veut un logiciel OpenSource
Scénario	Oui	L'utilisateur doit pouvoir créer ses propres scénarios
Retour d'états possible	Oui	L'utilisateur doit avoir un retour d'état lorsqu'il commande un équipement
Commande voix / texte	Commande depuis l'application	Transmettre les commandes via une commande texte

Tableau 15 : Comparaison entre Tahoma et la ZenBox

8.3.8. Fibaro

C'est le logiciel propriétaire utilisé par la box Home Center 2.

	FIBARO	ZenBox
Multiplateforme	serveur web + Android et iOS	Application Android et serveur web
Distance	Accès à distance depuis internet	Dans la maison mais on veut pouvoir par la suite communiquer depuis l'extérieur
Plusieurs comptes	Oui	Chaque utilisateur est différent et doit pouvoir se créer un compte pour gérer ses équipements

OpenSource / Propriétaire	Propriétaire	On veut un logiciel OpenSource
Scénario	Oui	L'utilisateur doit pouvoir créer ses propres scénarios
Retour d'états possible	Oui	L'utilisateur doit avoir un retour d'état lorsqu'il commande un équipement
Commande voix / texte	Commande vocale possible et commande depuis l'application	Transmettre les commandes via une commande texte

Tableau 16 : Comparaison entre Fibaro et la ZenBox

8.4. Réseau de neurones

Toutes les données de la maison sont récupérées et traitées par un algorithme qui agit comme un réseau de neurones afin de pouvoir apprendre et agir intelligemment.

Le **réseau de neurones** se rapporte à l'intelligence artificielle. Les humains s'inspirent du cerveau humain pour les créer. Le but est de reproduire le réseau de neurones du cerveau. C'est en 1943 que les premières notions de neurone formel apparaissent. Le neurone formel est une représentation mathématique et informatique d'un neurone biologique. [15] & [16]

Le neurone formel possède la même structure qu'un neurone biologique, à savoir des entrées et des sorties. C'est une fonction à plusieurs variables et à valeurs réelles. Les neurones formels sont donc connectés entre eux afin de former un réseau de neurones. Des poids synaptiques sont appliqués à chacune des entrées pour pondérer les valeurs, cela visant à faire fonctionner le neurone comme nous voulons.

Vient ensuite la phase d'apprentissage. En donnant d'abord des exemples aux neurones, on peut retrouver l'action à faire en fonction des données en entrée. On peut trouver la solution de manière empirique ou théorique en fonction de la complexité de la tâche. Certains algorithmes d'apprentissages existent.

Chaque neurone peut apprendre un nombre limité de choses. En augmentant le nombre de neurones dans le réseau alors on augmente la capacité d'apprentissage. L'algorithme d'apprentissage devient de plus en plus complexe en fonction du nombre de neurones dans le réseau.

8.5. Logiciels développant les réseaux de neurones

Certains logiciels existent déjà permettant de développer un réseau de neurones. C'est le cas de **TensorFlow** : c'est un logiciel Opensource développé initialement par l'équipe Google dans le but de développer et d'entraîner des réseaux de neurones. Il permet à Google de traiter toutes ses données afin d'améliorer le processus d'apprentissage de ses IAs. Ce logiciel est sorti en novembre 2015. [17] & [18]

On peut aussi trouver dans le même domaine **Deeplearning4j** : c'est un logiciel open source codé en JAVA qui permet aussi le développement des réseaux de neurones basé sur le cloud [33] ou encore **Thenao**: Logiciel toujours open source en python cette fois qui a un système appelé Blocks pour l'entraînement de réseau de neurones [34]. **Torch** : Un autre logiciel open source permettant le développement de réseau de neurone qui a la particularité de proposer un environnement proche de MatLab. [36]

Les réseaux de neurones sont une partie importante de notre projet puisqu'il s'agit de la partie qui permettra à notre box de comprendre les commandes envoyées par l'utilisateur sous forme de texte. Dans le cadre de notre projet nous allons ainsi être amené à choisir un logiciel de deeplearning open-source. Ce sont des logiciels qui permettent la création et l'entraînement de réseau de neurones. Dans le tableau ci-dessous, nous avons cherché des informations sur ces différents logiciels.

Nom	Langage	Créateur	Reconnaisse le texte ?	Interface	Open-source?
TensorFlow	Python, C++	Google	Oui	Python	Oui
CNTK	C++	Microsoft	Oui		Oui
Thenao (Blocks)	Python	Alexey Kryukov	Oui	Python	Oui
Torch	C++, Lua, LuaJIT, C, CUDA	Ronan Collobert, Koray Kavukcuoglu, Clement Farabet	Oui	Lua, LuaJIT, C	Oui
Deeplearning4j	C++, java	SkyMind	Oui	Java, Scala, C, C++, Python, Clojure	Oui
Caffe2	C++, python	Facebook	Oui	Python, MatLab	Oui
MXnet	C++	Communauté d'apprentissage machine	Oui	C++, Julia, Python, Matlab, R, Javascript, Go, Scala, Perl	Oui
OPEN NLP	Java	Apache	Oui	Java	Oui
CORE NLP	Java	Stanford University	Oui	Javascript, Python, Java, Clojure, Docker, Elixir, Lua, Perl, Python	Oui

Tableau 17 : Comparaison des différents logiciels développant des réseaux de neurones

Il existe encore beaucoup d'autre réseau de neurone que l'on peut entraîner et qui sont eux aussi open source. D'après les données que nous avons pu trouver nous serions plus intéressés par l'utilisation de Core NLP du fait que son interface fonctionne en Java et que c'est le langage dans lequel nous sommes le plus à l'aise. De plus il est possible de l'utiliser dans la plupart des langages de programmation existant ce qui nous laisse la possibilité de l'utiliser en JavaScript.

8.6. Différents protocoles de transmissions existants

Les fréquences sur lesquelles on peut émettre des signaux sont régies par la loi en France. Dans le domaine de la domotique, on retrouve les fréquences suivantes : 433 Mhz, 868Mhz, et 2.4Ghz. Voici différents protocoles que l'on peut retrouver en fonction des fréquences :

- 433Mhz: RTS, Oregon, X10
- 868 Mhz: Z-Wave, EnOcean, io-homecontrol, KNX
- 2.4Ghz : Zigbee, Bluetooth, Wifi

Nous allons en présenter quelques-uns dans la partie qui suit.

8.6.1. Zwave

Ce protocole permet la communication entre plusieurs appareils électroniques à l'aide des radios fréquences. Les informations peuvent être de toutes sortes, c'est-à-dire relevé de température ou actions (On/Off). Ce signal est utilisé dans les résidences classiques et a une portée de 30 mètres. Utilisé dans la domotique, ce protocole utilise un réseau maillé. Lors de la mise en marche du réseau, tous les éléments sont mis en relations. On envoie un ordre à un objet qui le renvoie à un autre objet et ainsi de suite jusqu'à atteindre la vraie destination. Pour cela, tous les composants sont constamment sur écoute. Ainsi, aucune architecture n'est définie, elle se construit au fur et à mesure. En revanche cela peut poser problème lorsque l'on ajoute ou supprime un composant. Il faut toujours resynchroniser pour que les nouveaux chemins se créent. Dans un réseau Z-Wave, il y a plusieurs types d'éléments, certains sont reliés au courant et d'autres avec des piles. De plus, ils sont tous émetteur et récepteur. C'est-à-dire qu'ils sont capables d'envoyer des informations (états, ordres) mais aussi d'en recevoir. On peut en mettre 232 dans un réseau. On peut créer plusieurs réseaux Z-Wave sans qu'ils interagissent entre eux car chaque réseau a un code qui lui est attribué. On utilise la modulation de signal FSK (Frequency shift keying). Afin d'accéder au Z-Wave il « suffit » à un composant d'embarquer une puce Z-Wave sur son circuit électronique. Dans cette puce, il y a un récepteur/émetteur RF, un microprocesseur, 32kB de mémoire flash contenant le protocole Z-Wave, l'application pour le gérer et certaines caractéristiques du réseau le concernant, des interfaces systèmes, un moteur 3DES et un contrôleur Triac. [3]

8.6.2. RFXcom

Ce produit permet à de nombreuses technologies domotiques de recevoir les trames RF 433Mhz.

RFXCOM permet de capter les émetteurs X10RF et les sondes Oregon Scientific. De plus, il peut aussi recevoir des codes de télécommandes ou des interrupteurs.

Pour gérer le RFXCOM, il faut le brancher en USB sur le PC. On va ainsi pouvoir enregistrer des températures par exemple.

RFXCOM propose un récepteur permettant de reproduire les actions de certaines télécommandes à partir de l'ordinateur.

Le RFXCOM peut être relié au port USB d'un Raspberry Pi.

8.6.3. RFLink

Rflink c'est une passerelle qui permet d'écouter les informations venant de divers modules radios (433mhz, 868mhz et 2.4ghz) mais aussi de leur donner des ordres

Au niveau des gateway, RFLink utilise un Arduino Mega, celui-ci doit être branché en USB sur un Jeedom (maître ou esclave). Il s'agit d'une copie de RFXcom en version open source. [32]

8.6.4. EnOcean

Cette technologie est leader mondiale en termes de collecte d'énergie sans fil. Le nouveau nom de cette technologie est : "Dolphin - Auto-powered IoT by EnOcean".

Elle permet l'automatisation des bâtiments en utilisant la bande inférieure à 1 GHz « 868 MHz EnOcean » pour l'Europe. Elle est également compatible avec les systèmes Zigbee et BLE. [5]

8.6.5. xPL

C'est un protocole qu'on utilise dans la communication en domotique. Les différents équipements de la maison échangent des messages bien définis. Ce protocole est réputé « d'assez facile » à utiliser et se base sur le protocole UDP. Cependant depuis septembre 2016, ce protocole a été abandonné. [8]

8.6.6. Zigbee

Ce protocole permet la communication de petites radios basée sur la norme IEEE 802.15.4 (système en couches) pour les réseaux personnels. Cette communication se fait surtout à courte distance. Le rayon d'action est faible mais très fiable. Ce système fonctionne avec un réseau maillé. C'est-à-dire que la configuration se fait automatiquement et est intelligente. Les nœuds du réseau fonctionnent durant plusieurs mois grâce à une pile 1.5V. Si on souhaite se connecter au réseau Zigbee la meilleure solution est de reprendre un produit développé et ensuite le modifier. Le routage entre les dispositifs est soit direct, soit indirect. Cela dépend si le dispositif connaît l'adresse du destinataire ou non. Tout cela est stocké dans une table de routage. [4]



Figure 8 : Logo Zigbee

8.6.7. IO Homecontrol

C'est une technologie utilisée par plusieurs grandes marques. Les derniers équipements motorisés de Somfy l'utilisent. [20]

Ce protocole est partagé par plusieurs grandes marques et rend possible la compatibilité entre différents équipements. C'est une technologie de radio sans fil. C'est un langage commun qui est utilisé et qui permet cette compatibilité. Niveau sécurité, la technologie scanne trois fréquences radios entre 868 et 870Mhz et envoie l'information sur celle qui est disponible. Les échanges sont assurés par une communication radio à saut de fréquences. C'est-à-dire que la communication change la fréquence des porteuses régulièrement selon un modèle connu de l'émetteur et du récepteur. Cela permet de rendre la communication plus difficilement détectable, et rend le signal plus résistant aux interférences. [20]

Cette technologie a été inventée en 1941 par Hedy Lamarr.

Une clé de cryptage est également utilisée afin de vérifier l'origine de chaque commande dans la technologie io-homecontrol. La technologie assure une communication dans un rayon de 20 mètres dans une construction en béton et 300 mètres en extérieur.

Il vaut mieux envisager cette technologie si on envisage de faire évoluer notre système domotique.

8.6.8. RTS (Radio Technology Somfy)

C'est le protocole historique de Somfy, qui est plus vieux. Cette technologie est disponible en grande surface. Il est propre à Somfy. Il est conseillé d'utiliser cette technologie lorsque l'on ne veut contrôler que quelques équipements motorisés et que l'on ne souhaite pas en rajouter trop par la suite.

Elle permet de piloter moins d'équipements que io-homecontrol puisqu'elle ne rassemble qu'une gamme d'équipements, contrairement à l'autre technologie. Il n'y a pas de retour d'information sur la commande en général (on ne peut pas savoir si notre volet est fermé ou non à distance par exemple, on ne peut que donner l'ordre de le monter ou de le baisser). [21]

8.7. Les équipements de la maison

Comme équipements présents dans cette maison, il y a les ampoules connectées. [1] & [2]

La plus présente sur le marché est la **Phillips Hue**. Cette ampoule est la plus puissante des ampoules connectées. Elle fonctionne sur le réseau Zigbee, ce qui permet de modifier l'éclairage à l'aide d'une connexion sans fil. Elle possède la programmation horaire, le réveil et la synchronisation de plusieurs éléments de la maison. De plus, on peut régler la luminosité en fonction de la situation. Par exemple il existe un mode concentration, mis au point par des experts, qui limite la teinte des lumières.

De plus, pour régler la teinte des lumières, il est possible d'utiliser les photos de l'utilisateur pour régler la couleur.

Les ampoules sont connectées via un Smartphone, cela se passe au niveau du pont Hue. On peut ainsi connecter 50 ampoules. Le fait de pouvoir régler ses lumières à distances permet de commander son éclairage et ainsi de faire des économies ou alors faire croire qu'il y a du monde de présent à la maison.



Figure 9 : Ampoule Phillips Hue

Il y a également les **Ikea Trådfri** :



Figure 10 : Ikea Trådfri

Cet ampoule créée par Ikea permet de régler sa lumière en fonction de son humeur. Cet lampe est couplée avec un variateur qui permet, à l'aide du bout des doigts, de diminuer ou augmenter l'intensité de la lumière. De plus, certaines améliorations de cette lampe permettent de changer la couleur de la lumière.

Parmi les équipements domotiques, on peut trouver des thermostats connectés : **[6]**
Nest : Le thermostat Nest est un thermostat intelligent. Il utilise le langage OpenTherm qui permet aux thermostats de communiquer entre eux afin de contrôler la production d'eau chaude et de chauffage. Il permet ainsi de faire de nombreuses économies d'énergie en gérant les périodes de chauffe du ballon d'eau chaude. Ces périodes sont définies à l'aide d'un auto apprentissage, en tenant compte de l'occupation de la maison, des conditions météorologiques.

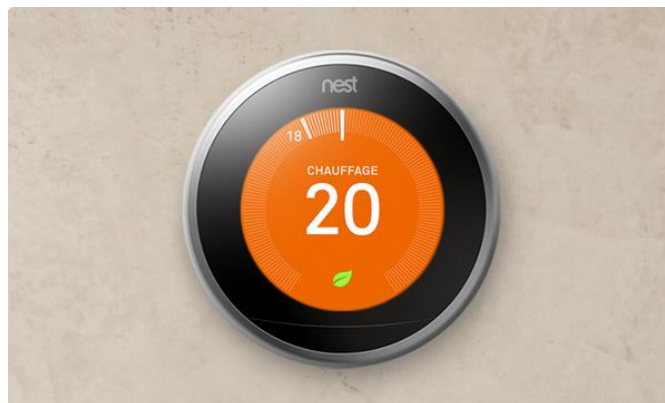


Figure 11 : Thermostat Nest

Ecobee : Il fonctionne avec des capteurs positionnés dans la maison qui lui permettent de détecter la présence ou non de personnes afin de réguler le chauffage en fonction. Il possède également une reconnaissance vocale permettant de se coupler avec Amazon Alexa



Figure 12 : Thermostat Ecobee

Il existe aussi des solutions pour commander des volets, pour cela il faut remplacer la prise murale par un interrupteur télécommandé pour volet et store.

Après avoir fait l'inventaire des solutions existantes sur le marché, on peut voir que notre solution n'existe pas encore. En effet, aucun produit du marché ne propose une solution ne stockant pas ses données sur le cloud et compatible avec la plupart des protocoles de transmission. La recherche sur les réseaux de neurones nous a permis de dégager le logiciel qui nous semblait le plus adapté à nos besoins afin de développer notre solution.

9. Conclusion

Nous avons défini dans ce premier livrable plusieurs choses essentielles au bon déroulement de notre projet. La présentation de l'équipe et du mode de fonctionnement de celle-ci nous a permis de poser des bases d'organisation qui nous permettront d'être plus efficace par la suite. Puis nous avons défini un cahier des charges le plus précis possible nous donnant des objectifs clairs à atteindre. Cela nous permet d'avoir une ligne directrice tout au long de notre projet. Nous nous sommes appuyés sur ce cahier des charges dans l'état de l'art pour comparer les solutions existantes à notre proposition, et cela nous a permis de montrer que notre solution n'existait pas encore à l'heure actuelle sur le marché. La recherche d'outils qui nous permettront de mener à bien notre projet nous a déjà permis de dégager les logiciels qui nous seront utiles en fonctions de nos besoins.

Ce projet, étalé sur l'année scolaire entière, va nous demander d'être bien organisés tout au long pour pouvoir le mener à bien. Ce premier livrable pose des bases solides, et nous pourrons nous y référer si nous nous sentons perdus et éventuellement venir modifier les objectifs si besoin est.

Bibliographie

- [1] «Lumière connectée Hue,» [En ligne]. Available: <https://www.philips.fr/c-p/8718291241751/hue-lumiere-connectee-hue/sp%C3%A9cifications>.
- [2] «Comparatif des ampoules connectées,» [En ligne]. Available: <https://www.objetconnecte.net/comparatifs/guide-et-comparatif-ampoule-connectee/>.
- [3] «Z-Wave,» [En ligne]. Available: <http://www.domotique-info.fr/technologies-domotique/zwave/>.
- [4] «Zigbee - Domotique info,» [En ligne]. Available: <http://www.domotique-info.fr/technologies-domotique/zigbee/>.
- [5] «Energy Harvesting Wireless Sensors Solutions,» [En ligne]. Available: <https://www.enocean.com/en/>.
- [6] «Qu'est ce que la technologie Opentherm,» [En ligne]. Available: <https://nest.com/fr/support/article/What-is-OpenTherm-and-what-can-the-3rd-generation-Nest-Learning-Thermostat-do-with-it>.
- [7] «Jeedom Interactions,» [En ligne]. Available: <https://guillaumebraillon.fr/jeedom-interactions-gestion-des-lumieres/>.
- [8] «The xPL Project,» [En ligne]. Available: <http://xplproject.org.uk/>.
- [9] «Présentation du RFXcom,» [En ligne]. Available: <http://www.planete-domotique.com/rfxcom/>.
- [10] «Alexa,» [En ligne]. Available: <https://developer.amazon.com/alexa>.
- [11] «Amazon Echo,» [En ligne]. Available: <http://www.journaldunet.com/ebusiness/internet-mobile/1194132-amazon-presente-3-nouvelles-versions-de-son-haut-parleur-echo/>.
- [12] «Amazon Echo bis,» [En ligne]. Available: <https://www.objetconnecte.net/amazon-echo/>.
- [13] «Achetez Google Home,» [En ligne]. Available: <https://store.google.com/>.
- [14] «Homepod Apple,» [En ligne]. Available: <https://www.apple.com/homepod/>.
- [15] «Réseaux de neurones,» [En ligne]. Available: <https://www.math.univ-toulouse.fr/~besse/Wikistat/pdf/st-m-app-rn.pdf>.
- [16] «Réseaux de neurones bis,» [En ligne]. Available: <http://www.grappa.univ-lille3.fr/polys/apprentissage/sortie005.html>.
- [17] «TensorFlow,» [En ligne]. Available: <https://www.tensorflow.org/>.
- [18] «TensorFlow in 5 Minutes,» [En ligne]. Available: <https://www.youtube.com/watch?v=2FmcHiLCwTU>.

- [19] «Accueil,» [En ligne]. Available: <http://www.io-homecontrol.com/index.php/fr/>.
- [20] «Étalement de spectre par saut de fréquence,» [En ligne]. Available: https://www.google.fr/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0ahUKEwju_O-JzurWAhXJqxoKHVP6ABoQFggNMAA&url=http%3A%2F%2Fwww.institutmaupertuis.fr%2Finclude%2Ftelechargement.php%3Fid_doc%3D89%26fichier%3D1&usg=AOvVaw0SmiykTl6oKyLQGtsFI-dp.
- [21] «RTS ou IO Homecontrol,» [En ligne]. Available: <https://www.kalytea.com/img/cms/RTS%20ou%20IO%20homecontrol%20Somfy.pdf>.
- [22] «domoticz,» [En ligne]. Available: <http://www.domoticz.com/>.
- [23] «Home Yadoms,» [En ligne]. Available: <https://github.com/Yadoms/yadoms/wiki>.
- [24] «Calaos, Domotique libre,» [En ligne]. Available: <https://calaos.fr/fr/>.
- [25] «Home Assistant,» [En ligne]. Available: <https://home-assistant.io/>.
- [26] «OpenHAB,» [En ligne]. Available: <http://www.openhab.org/>.
- [27] «Jeedom - Les box,» [En ligne]. Available: <https://www.jeedom.com/site/fr/box.html>.
- [28] «Qu'est ce que c'est - eedomus,» [En ligne]. Available: <http://www.eedomus.com/fr/quest-ce-que-cest/#top>.
- [29] «Home Center 2,» [En ligne]. Available: <https://www.fibaro.com/fr/products/home-center-2/>.
- [30] «Centrale domotique Home Center 2,» [En ligne]. Available: <http://www.planete-domotique.com/centrale-domotique-home-center-2-fibaro.html>.
- [31] «Box domotique Tahoma,» [En ligne]. Available: https://www.somfy.fr/produits/domotique/maison-connectee-tahoma?gclid=CjwKCAjwpfzOBRA5EiwAU0ccNzGuyG2T5r5b2X7KsJPwvHaKC44BiVGsJIN0Upqt4FZoja9df6RgRoCElQQAvD_BwE.
- [32] «Appliances Sciences : Alexa,» [En ligne]. Available: <https://www.cnet.com/news/appliance-science-alexa-how-does-alexa-work-the-science-of-amazons-echo/>.
- [33] «RFLink Development,» [En ligne]. Available: <http://www.rflink.nl/blog2/development>.
- [34] «Qu'est ce que deeplearning4,» [En ligne]. Available: <https://deeplearning4j.org/fr-index.html>.
- [35] «Theano,» [En ligne]. Available: <http://deeplearning.net/software/theano/>.
- [36] «SoftwareLink,» [En ligne]. Available: http://deeplearning.net/software_links/.

Annexe :

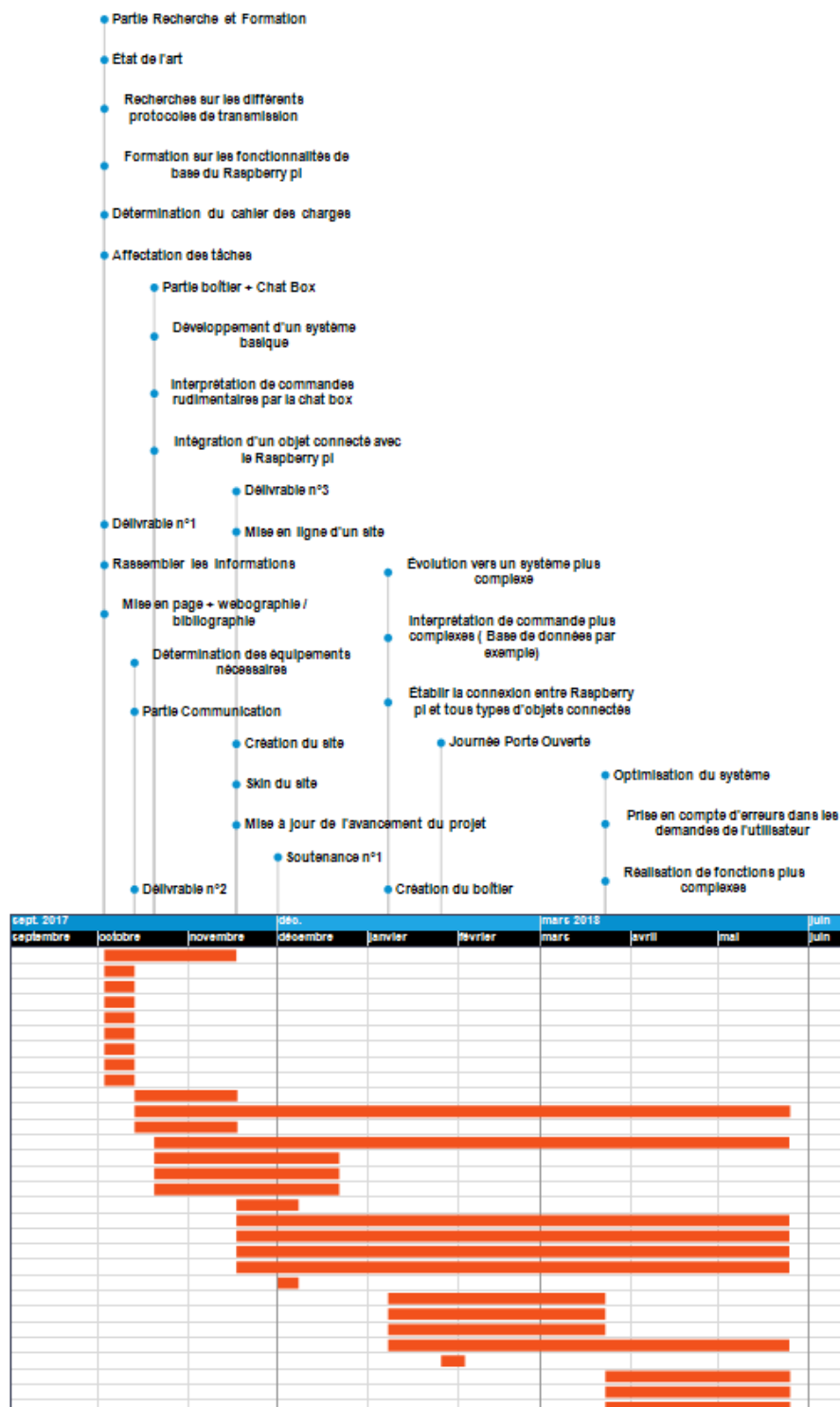


Figure 13 : Diagramme de Gantt