

CoBox

Github : github.com/hackersoft13/Cobox

Site web : cobox.paulitow.fr

RAPPORT D'ACTIVITE

3^{ème} *itération*



Projet de système d'exploitation
Moyse – Desmarets – Ilhe - Legeas

ISEN | école
d'ingénieurs
ALL IS DIGITAL!

1. Rappel.....	2
1.1 Besoins	2
1.2. Contrainte	3
1.3. Roadmap	4
2. Avancés.....	5
2.1. 3^{ème} itération – 14/12/2019 12:30	5
2.1.1. IHM Web & communication avec BDD.....	5
2.1.2. Cobox Serveur – Multi clients & Pipes Nommés	6
2.2.1. Documentation	7
2.2.2. Base de donnée	Erreur ! Signet non défini.
2.2.3. Cobox Serveur – Software	Erreur ! Signet non défini.
2.2.2. Indicateurs	9

I.1 Besoins

Le but du projet Cobox est de concevoir une solution de domotique pour particuliers, permettant le contrôle et la supervision de différents éléments (tout équipement pouvant être commandé par un relais) au sein d'un foyer.

Cette solution se décomposera en une centrale domotique, et X clients. Les clients étant les actionneurs / capteurs.

A ce jours les besoins concernant la centrale sont :

- Echanger des informations à ses actionneurs en mode serveur via Wifi directe (mode hotspot)
- Afficher de manière claire les mesures des différents éléments supervisés par la centrale

Et concernant la partie client :

- Echanger des informations à la centrale en mode client via Wifi directe
- Piloter les actionneurs via des GPIO*
- Acquisition de données via des capteurs (Température / humidité...)

I.2. Contrainte

Les contraintes de ce projet ont été définies en accord avec notre client et sont susceptibles d'être modifiées avec l'évolution du présent document.

Contraintes	
CENTRALE	La centrale doit comporter un système d'exploitation linux
	La centrale doit utiliser un programme développé en C pouvant dialoguer avec les différents actionneurs
ACTIONNEUR	L'actionneur doit être capable de dialoguer via Wifi
	L'actionneur doit être capable de piloter jusqu'à 8 relais
	L'actionneur doit être capable de monitorer un capteur et de transmettre sa valeur à la centrale

I.3. Roadmap

Ci-dessous, la Roadmap définissant les étapes cruciales de notre projet

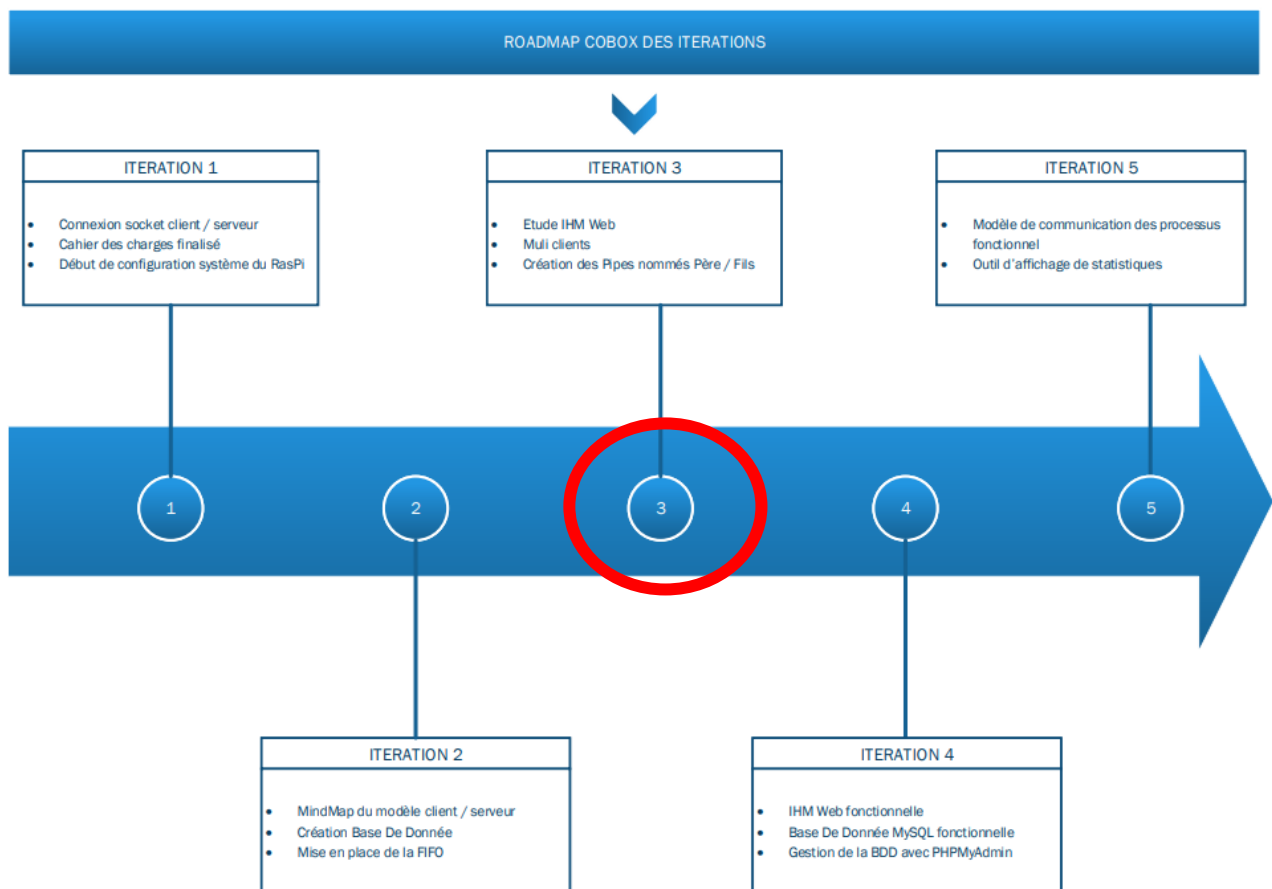


Figure 1 : Roadmap Cobox

Nous sommes donc à l'heure de la rédaction à l'itération n°4

2.1. 3^{ème} itération – 14/12/2019 12:30

Ci-dessous sont détaillées les avancées récentes sur notre projet

2.1.1. IHM Web & communication avec BDD

2.1.1.1. Effectif à ce jour

Nous avons opté pour un IHM XAMPP sous distribution Ubuntu comportant :

- Un Apache2 comme serveur HTTP
- Une base de données Maria DB récupérant
- Un moteur PHP qui permet la communication BDD / Web
- Une bibliothèque Perl qui ne sera pas utile

Une page Bootstrap HTML affichera le contenu de la base de données, sous forme d'indicateurs.

2.1.1.2. Prochaine étape

Les prochaines tâches à effectuer sont :

- Configuration de la base XAMPP
- Développement de la page Web
- Communication PHP fonctionnelle

2.1.2. Cobox Serveur – Multi clients & Pipes Nommés

La solution avec les pipes nommés a été remplacée par du **Shared Memory Segment** (SMS), qui transmet les données du fils vers le père.

Server.c

```
key_t key = ftok("shmfile",65);
int shmid = shmget(key,1024,0666|IPC_CREAT);
int sockfd, ret, i;
struct sockaddr_in serverAddr;
int newSocket;
struct sockaddr_in newAddr;
socklen_t addr_size;
char buffer[1024];
pid_t childpid;
measure *m=NULL;
m = (measure*)malloc(sizeof(measure));
i=0;
sockfd = socket(AF_INET, SOCK_STREAM, 0);
```

Composition de l'infrastructure à ce stade :

- 1 serveur PI
- 1 client ESP8266
- 1 DHT22
- 1 LED

Le tout communique ensemble.

A l'heure actuelle, les éléments suivants sont effectués :

- Conception et câblage physique du prototype
- Utilisation de la Raspberry comme serveur
- Implantation des mesures en BDD.
- Squelette d'une IHM

Il reste donc :

- Communication des valeurs à la BDD

2.2. Documentation

Vous pouvez retrouver la MindMap du fonctionnement global de notre infrastructure Cobox serveur ci-dessous :

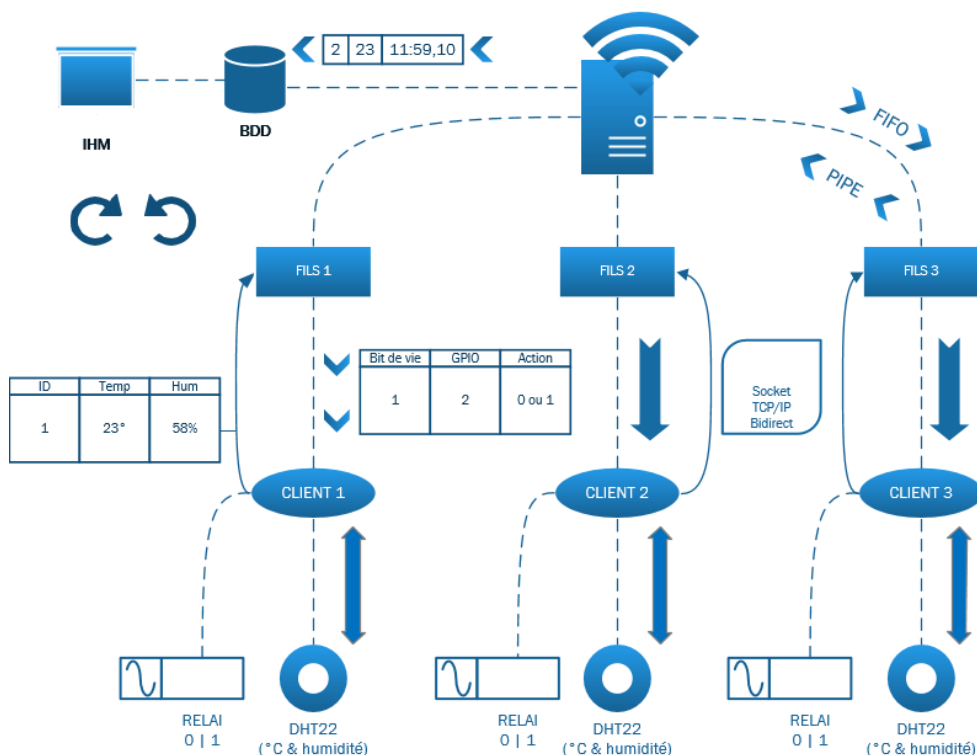


Figure 2 : Synoptique de fonctionnement

2.2.2. Indicateurs

